Toward Enabled Industrial Verticals in 5G: A Survey on MEC-Based Approaches to Provisioning and Flexibility

Francesco Spinelli and Vincenzo Mancuso, *Member, IEEE*

***Abstract*—The increasing number of heterogeneous devices connected to the Internet, together with tight 5G requirements have generated new challenges for designing network infrastructures. Industrial verticals such as automotive, smart city and eHealthcare (among others) need secure, low latency and reliable communications. To meet these stringent requirements, computing resources have to be moved closer to the user, from the core to the edge of the network. In this context, ETSI standardized Multi-Access Edge Computing (MEC). However, due to the cost of resources, MEC provisioning has to be carefully designed and evaluated. This survey firstly overviews standards, with particular emphasis on 5G and virtualization of network functions, then it addresses flexibility of MEC smart resource deployment and its migration capabilities. This survey explores how the MEC is used and how it will enable industrial verticals.**

***Index Terms*—Multi-access edge computing, ETSI, resources migration, NFV, 5G, MEC provisioning, industrial verticals, automotive, IoT, smart city, video streaming, AR/VR applications, smart factory, eHealthcare.**

I. Introduction

NOWADAYS, two main paradigms are changing the design of the network infrastructure: Cloud Computing and Network Function Virtualization (NFV). Their main features consists of the use of a shared pool of computing resource (Cloud Computing) [1] and the softwarization of hardware functions (NFV) [2].

The concept of Multi-Access Edge Computing (MEC) embeds the two above-listed paradigms, which is vital for the success of industrial *verticals* such as IoT, automotive, augmented/virtual reality (AR/VR), eHealthcare, Entertainment, smart factories, and smart cities [3], [4]. All these verticals need different and stringent requirements in order to achieve both an optimal Quality of Experience (QoE) and Quality of Service (QoS), and independent Cloud Computing, Software Defined Networking (SDN) and NFV mechanisms are not necessarily sufficient to guarantee them. For instance, AR/VR applications need to have very low delays, difficult to achieve if the servers are located in a data center thousands of kilometers away from end users. Therefore, computing resources need to be placed at the edge of the network and orchestrated with network resources and functions. The MEC, however, is not the only solution proposed in the last years. One other example is the Fog Computing paradigm, a solution supported by the Open Fog Consortium.[1] With Fog Computing, the computing resources are placed between the core and the edge of the network, allowing therefore to obtain better performances with devices that need, for instance, low latency and mobility support [5]. A possible example of Fog Computing application is Cloudlet, which is a cluster of computers directly accessible from the mobile devices and connected to the Internet [6]. Anyway, the paradigm of MEC is different from Fog Computing, since it promises to bring resources really at the edge of the network, supporting several access technologies and generating new interesting scenarios for both vendors and developers.

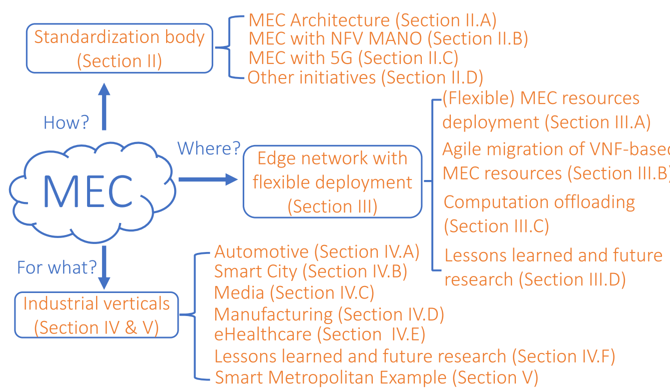

Fig. 1. The three main concepts evaluated in this survey.

A. Related Surveys

After its initial standardization in 2014, the MEC concept started to become popular among researchers and in the following years many early surveys appeared on the topic of

Manuscript received January 2, 2020; revised June 15, 2020 and November 5, 2020; accepted November 7, 2020. Date of publication November 16, 2020; date of current version February 24, 2021. This work was supported by the Region of Madrid through the TAPIR-CM Project under Grant S2018/TCS-4496. The work of Vincenzo Mancuso was supported by the Ramon y Cajal under Grant RYC-2014-16285. *(Corresponding author: Francesco Spinelli.)*

Francesco Spinelli is with the Opportunistic Architectures Lab, IMDEA Networks Institute, 28918 Madrid, Spain, and also with the Department of Telematics, Universidad Carlos III of Madrid, 28903 Getafe, Spain (e-mail: francesco.spinelli@imdea.org).

Vincenzo Mancuso is with the Opportunistic Architectures Lab, IMDEA Networks Institute, 28918 Madrid, Spain (e-mail: vincenzo.mancuso@imdea.org).

Digital Object Identifier 10.1109/COMST.2020.3037674

[1] https://www.openfogconsortium.org/

TABLE I
RECENTLY APPEARED SURVEYS FOCUSING ON MEC

Title	Focus	Year
On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration [7]	It focuses on MEC orchestration and on the different deployment scenarios and tools	2017
Game theory for multi-access edge computing: Survey, use cases, and future trends [8]	It discusses the use of Game Theory with MEC resources and how to balance diverse tradeoffs	2019
A survey on mobile edge networks: Convergence of computing, caching and communications [9]	A comprehensive survey regarding issues on computing, caching, and communication techniques at the network edge	2017
A survey on the edge computing for the Internet of things [10]	The survey analyzes how edge computing improves the performance of IoT networks, focusing on architecture and KPI such as latency, bandwidth occupation, energy consumption.	2018
How can edge computing benefit from software-defined networking: A survey, use cases, and future directions [11]	In this survey, the authors show how MEC can leverage SDN	2017
Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges [12]	It analyzes security threats and challenges typical of Edge Computing environments	2018
Survey on multi-access edge computing for Internet of things realization [13]	It provides an overview of MEC exploitation for the realization of IoT applications and discusses the technical aspects of enabling MEC in IoT	2018
The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective [14]	It focuses on MEC models, architectures and approaches for security and resilience	2017
A survey of fog computing: Concepts, applications and issues [5]	It discusses the definition of fog computing and similar concepts	2015
A survey on mobile edge computing: The communication perspective [15]	This survey deals with joint radio-and-computational resource management with MEC	2017
Multi-access edge computing: open issues, challenges and future perspectives [16]	It reviews the prevalent Edge Cloud Computing frameworks	2017
Mobile Edge Computing: A Survey on Architecture and Computation Offloading [17]	This survey focuses mainly on computation offloading leveraging on MEC resources	2017
A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art [18]	It delineates the integration of MEC with 5G and its new technologies	2019

deploying computing and networking within the same infrastructure. Table I reports the most extensive surveys appeared since MEC was defined.

The first survey that started to address MEC-like topics is from Yi *et al.* [5]. It was published in 2015 and describes both the Edge and Fog computing paradigms, even though the main focus was on the latter. Later, more surveys on MEC appeared. In [15], Mao *et al.* give a comprehensive view of the paradigm, focusing especially on the joint radio and computational resource management. Moreover, in that paper, the authors identified potential future research challenges on MEC deployment, cache-enabled MEC, mobility management for MEC, green MEC, etc., some of which have been later investigated. Roman *et al.* [12] give an holistic perspective of all the security threats affecting the edge paradigms (considering hence also MEC). Instead, Shirazi *et al.* [14] focuses on security and resilience-related mechanisms such as anomaly detection and policy-based resilience management in the cloud, arguing that the MEC paradigm would suit their implementation. Wang *et al.* [9] deal with the most important issues on computing, caching, and communication techniques at the network edge. Taleb *et al.* [7] focus on different MEC key enabler technologies. Their paper ranges from the reference architecture provided by ETSI [4] to the MEC service and network orchestration scenarios (considering MEC service mobility and joint optimization of VNFs and MEC services) ending with an overview of MEC deployment issues. In [16], Shahzadi *et al.* provides a review of the principal edge cloud computing frameworks and approaches, evaluating them on the basis of QoS metrics. Baktir *et al.* [11] describe how MEC can leverage SDN mechanisms. On a different line, the papers of Yu *et al.* [10] and Porambage *et al.* [13] deal with positioning MEC in the IoT industry. In [17], Mach and Becvar focus on computation offloading leveraging MEC. They classify research efforts according to three patterns: decision on computation offloading, allocation of computing resource within the MEC, and mobility management. More recently, Pham *et al.* [18] overview the topic of MEC integration with 5G and new technologies such as Non-Orthogonal Multiple Access (NOMA), Energy Harvesting (EH), Wireless Power Transfer (WPT), UAV Communications, IoT, Heterogeneous Cloud Radio Access Network and Machine Learning. On a more theoretical front, Moura and Hutchison [8] overview the use of game theory for MEC, focusing especially on optimizing resource-constrained systems and balancing diverse tradeoffs. All the existing surveys provide valuable contribution, although none of the existing works provide a comprehensive view of how the MEC is evolving and is permeating vertical industry frameworks. This is happening thanks to the MEC versatility in terms of resource provisioning and flexibility, whose analysis is one of the key novel aspects of our work.

B. Novel Contribution

More in detail, Fig. 1 illustrates the three main concepts analyzed in this survey (standardization, flexible deployment, and application to verticals), for which we provide four main pieces of contribution:

- In order to make the MEC diffusion easier, avoiding any compatibility issues between different network providers, MEC needs to be fully standardized. This survey, therefore, presents a clear up-to-date comprehensive description of all the most important ETSI standardization contributions to MEC, focusing especially on (*i*) its reference architecture, (*ii*) how MEC leverages the ETSI NFV MANO framework, and (*iii*) merges

TABLE II
SUMMARY OF MAIN ACRONYMS

Acronym	Definition	Acronym	Definition
AF	Application Function	M2M	Machine to Machine
AP	Access Point	NEF	Network Exposure Function
AR/VR	Augmented Reality/Virtual Reality	NFV	Network Function Virtualization
CAM	Cooperative Awareness Messages	NFVI	NFV Infrastructure
CAPEX	Capital expenditure	NFVO	NFV Orchestrator
CDN	Content Delivery Network	NOMA	Non-Orthogonal Multiple Access
CFS	Customer Facing Service	OPEX	Operating expense
CPS	Cyber Physical Systems	OSS	Operations Support System
C-V2X	Cellular Vechicle to Everything	QoE	Quality of Experience
DENM	Decentralized Environmental Notification Message	QoS	Quality of Service
DL	Downlink	RAN	Radio Access Network
DNN	Deep Neural Network	RNI	Radio Network Information
D2D	Device to Device	SDN	Software Defined Network
EH	Energy Harvesting	SDO	Standard Development Organization
ETSI	European Telecommunications Standards Institute	UAV	Unmanned Aerial Vehicle
FMeC	Follow Me edge Cloud	UE	User Equipment
LADN	Local Area Data Network	UL	Uplink
LISP	Locator/Identifier Separation Protocol	URLLC	Ultra Reliable Low Latency Communications
ICN	Information Centric Network	UPF	User Plane Function
IIoT	Industrial Internet of Things	VIM	Virtualization Infrastructure Manager
IoT	Internet of Things	VNF	Virtual Network Function
ISG	Industry Specification Group	VNFM	Virtual Network Function Manager
ITS	Intelligent Transport Systems	VRU	Vulnerable Road User
MANO	NFV Management and Network Orchestration	V2I	Vehicle to Infrastructure
MDP	Markov Decision Process	V2V	Vehicle to Vehicle
MEAO	Mobile Edge Application Orchestrator	V2X	Vehicle to Everything
MEO	Multi-access Edge Orchestrator	WPT	Wireless Power Transfer
MEPM-V	MEC Platform-NFV	5GC	5G Core network
ML	Machine Learning	5GAA	5G Automotive Association
mmWave	Millimeter Wave	5G ACIA	5G Alliance for Connected Industries and Automation

with 5G networks. The survey also highlights (*iv*) other MEC-relevant ETSI standardization efforts (MEC pairing with C-RAN, MEC deployment in enterprise settings, MEC support for Network Slicing etc.) together with several attempts by other organizations, companies or researchers. Other surveys have partially covered this topic. References [7], [14], [15], [17] cover the general ETSI MEC framework while [18] covers the ETSI MEC infrastructure as it is supposed to work in 5G networks. However, our survey brings a fresh and novel cutting-edge comprehensive summary of all the ETSI MEC, 3GPP and works-related standardization efforts regarding the MEC.

- The core of the survey focuses on MEC provisioning features, with the goal of answering the question *Where should MEC resources be deployed?*. The two key features are: (*i*) flexibility of MEC resources deployment, where the flexibility is given by the integration with NFV, and (*ii*) agile migration of Virtual Network Functions (VNFs) across the edge infrastructure. The latter is a crucial point for MEC, because it gives the possibility to cope with user mobility and to have an optimal use of the possibly scarce edge resources. Furthermore, these features enable novel solutions for computation offloading, which is a well-known MEC research problem [17]. Within this context, the survey provides a detailed overview of the most relevant and recent papers in the area, together with identifying challenges and possible future directions. While it is true that this section describes two topics partially addressed as future work in previous publications (see [7], [15], [16], [17]), our survey provides an updated overview and compares what has been done and what still is an open challenge for researchers and developers.
- The survey highlights the state-of-the-art on the MEC integration in several industrial verticals defined by 5G-PPP[2] (i.e., automotive, smart city, media, manufacturing and eHealthcare). Existing challenges and lessons learned are discussed. This survey is the first one in proposing such in-depth analysis of MEC in industrial verticals, while [9] and [18] only provided high-level considerations on the topic.
- Another piece of contribution that we develop in the final part of the manuscript is the study of MEC deployment bottlenecks and costs in a smart metropolitan area, where disparate verticals can coexist. Specifically, we show that deploying new resources at the edge in a heterogeneous scenario has to be carefully studied and several bottlenecks should be taken into account for a successful MEC deployment. Even in the presence of quite powerful edge computing capabilities, we show how hard it is for the network infrastructure to support several verticals at the same time when the customer population becomes dense

C. Survey Organization

The remainder of this article is structured as follows. Section II overviews MEC standardization efforts. Section III focuses on two MEC key features: the flexible MEC resources

[2] https://5g-ppp.eu/verticals/

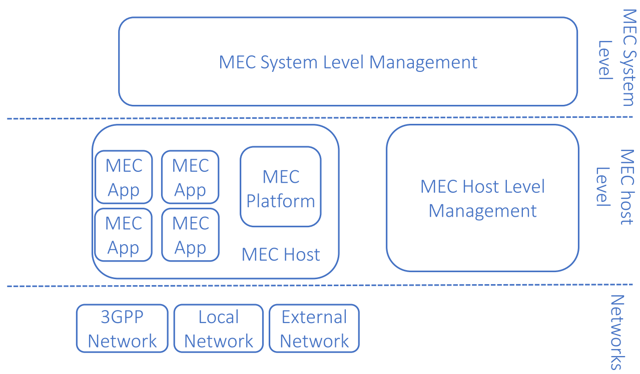

Fig. 2. ETSI MEC Framework.

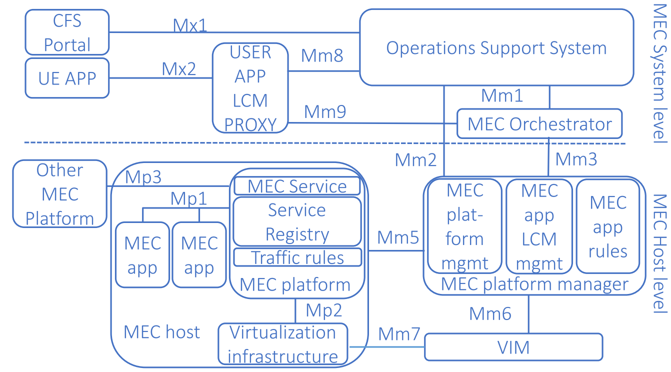

Fig. 3. MEC Architecture.

deployment and the agile migration of resources. Furthermore, it overviews the most recent works on computation offloading. Section IV describes the MEC support for several industrial verticals: automotive, smart city, eHealthcare, AR/VR applications and smart factories. Section V shows the potential of a MEC deployment in a smart metropolitan region. Finally Section VI summarizes the findings of our survey and concludes the work. Table II provides a summary of the main acronyms.

II. MEC Standardization

According to ETSI, MEC is *IT service environment and cloud-computing capabilities at the edge of the mobile network, within the Radio Access Network (RAN) and in close proximity to mobile subscribers* [3]. Examples of MEC applications include caching of DNS entries, caching of contents to deliver to customers, and tracking of devices. Furthermore, there are many proposals about using the MEC for implementing advanced network functions, e.g., for enhanced secure VPNs, as well as for computational offloading and collaborative computing purposes, indoor localization, distributed data analytics, assisted driving and control of vehicle platoons, smart infotainment with adaptive video transcoding and support for augmented and virtual reality, control of smart grids, and support for IoT and smart environments in general (smart cities, smart factories, smart health care systems, etc.).

MEC has been standardized by ETSI, which created a MEC Industry Specification Group (MEC ISG) and published the first white paper in September 2014 [19]. ETSI has released and updated more white papers and technical specifications in the following years. A new technology standardization process is very important for many reasons: *(i)* it allows to have interoperability between products, *(ii)* to brainstorm and clarify the challenging technical aspects and *(iii)* merge technical solutions together with research advancements.

Fig. 2 illustrates the general entities involved in the MEC architecture, according to ETSI [20]. Three different levels are present: the upper one is the MEC *System Level*, which has a global visibility on the MEC architecture and therefore coordinates every block in the levels below. In the middle, the MEC *host level* includes MEC host and MEC host level management. The MEC host is an entity that includes the platform and the virtualization infrastructure used to run the MEC, and which provides network resources, storage, MEC services and computing power for MEC applications. MEC services are provided and consumed by MEC applications or the MEC platform. Some examples are the Radio Network Information (RNI), which gives information on the radio network state, the location service, which gives location-related information and the bandwidth manager service, which helps in prioritize and handle traffic. Containers or virtual machines run as well in the MEC host and can leverage MEC services. At the bottom of the stack, Fig. 2 shows various transmission entities such as 3GPP cellular networks and local/external networks. This shows that the MEC will be able support many access technologies, even at the same time, giving the possibility to exploit fixed mobile convergence,[3] which is a 5G feature meant to allow devices to connect through both wired/wireless transmissions at the same time.

The reminder of this section is further divided into four different sub-sections: Section II-A highlights the main entities of a general MEC architecture while Section II-B and Section II-C highlight the MEC support for the ETSI NFV MANO architecture and for 5G networks, respectively. Finally, Section II-D overviews standardization efforts by other organizations and projects.

A. MEC Architecture

Figure 3 shows all the most important elements contained inside the MEC reference architecture, and the reference points connecting the whole system. Reference points are divided in 3 different categories:

- *Mp* are the reference points located inside a MEC platform, allowing the connectivity between MEC platforms, MEC applications and the data plane.
- *Mm* reference points are instead for management purposes.
- *Mx* reference points connect MEC elements towards external entities.

Describing the MEC system from the top (hence from the *system level*), requests to the MEC infrastructure are sent in two different ways: with a User Equipment (UE)/Device

[3]https://www.telefonica.com/en/web/press-office/-/telefonica-presents-the-first-prototype-of-an-open-and-convergent-access-network-that-integrates-fixed-and-mobile-and-enables-edge-computing

Application, or through a Customer Facing Service (CFS) portal. The latter allows operators' third parties to select a set of MEC applications given their needs and it is directly connected to the Operations Support System (OSS) through the *Mx1* reference point. Instead, from the UE, the requests are first sent through a *Mx2* reference point to the User application life cycle management proxy. This entity checks if the requested application is already instantiated and, otherwise, it forwards the request to the OSS. Moreover, it also informs the UE about the state of the application and it supports applications relocation inside or outside the MEC system. It is connected to the OSS through the *Mm8* reference point, and to the Multi-Access Edge Orchestrator (MEO) via the *Mm9* reference point.

The OSS receives the requests from both the CFS and the proxy and determines request granting, sending the requests to the MEO in positive cases. The OSS leverages the *Mm1* reference point, which triggers the instantiation and the termination of MEC applications, and on the *Mm2* to connect with the MEC platform manager. Furthermore, the OSS gives the possibility, upon device request, to relocate MEC applications to external clouds. The last element of the *system Level* is the MEO. It maintains an overall view of the MEC system, knowing the available resources, services, deployed MEC hosts, and it also monitors the topology. It selects the best host where to deploy an application, taking into account available resources, services availability and constraints such as latency. Moreover, it is responsible for operator policies and it interfaces with the Virtualization Infrastructure Manager (VIM) for preparing the physical infrastructure. It is connected with the MEC platform manager via the *Mm3* reference point, for application life cycle management and for keeping track of the available MEC services, and with the VIM.

Three different entities are present in the MEC *host level*: the MEC host, the VIM and MEC platform manager. The latter is responsible for managing the life cycle of both applications and MEC platforms, and for receiving information on faults and performance measurements from the VIM, hence informing the MEO if any relevant event happens. The MEC platform manager is connected to the VIM via the *Mm6* reference point and to the MEC platform via *Mm5* reference point, allowing the platform configuration and applications life cycles procedures. The VIM allows the management of the virtualization infrastructure located inside the MEC host, managing the allocation and release of virtualized resources, preparing the infrastructure to run a software image and it supports the rapid provisioning of applications, as described in [21]. It is connected with the virtualization infrastructure through the *Mm7* reference point. The MEC host is further divided in three different sub-entities.

- Virtualization infrastructure, which provides the computing and network resources and the data plane.
- MEC platform, which offers its services to the applications and talks with other MEC platforms under the same MEO; moreover, the MEC platform receives traffic rules and DNS configurations from the MEC platform manager and instructs the data plane following those rules.
- MEC applications are deployed as virtual machines or containers on top of the virtualization infrastructure. They interact with the MEC platform, providing the required services or leveraging on already instantiated MEC services and management information. Services hence are placed either inside the MEC applications or in the MEC platform, meaning they are directly deployed and controlled during the MEC platform instantiation.

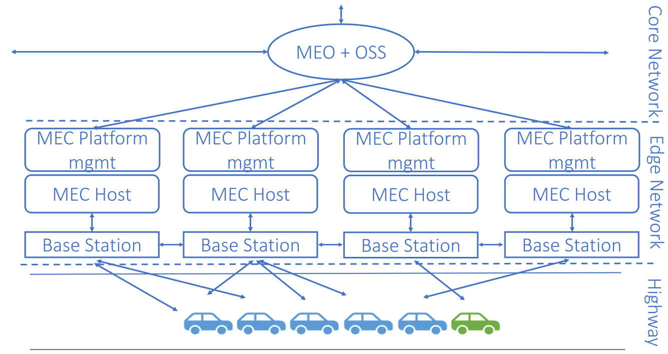

Fig. 4. MEC deployment in Platooning use case.

As an example, Fig. 4 shows a possible ETSI MEC framework deployment supporting platooning of assisted-driving vehicles. Thanks to its distributed architecture, the MEC host is deployed at the network edge, near the base station (the *gNB*, using the 5G jargon), while MEO and OSS, which need a more centralized view, can be deployed more inside the network. The MEC will provide support for platooning by storing, updating, processing and sharing information about road traffic, handling requests to join or leave the platoon, or helping vehicles by offloading part of their computation tasks.

B. MEC and NFV MANO

NFV is becoming more and more important since it enables operators to save money by transforming specialized and expensive hardware functions into software, therefore exploiting commodity hardware. ETSI standardized the NFV paradigm within the management and orchestration context since December 2014, in their NFV MANO project [22]. Afterwards, the authors in [23] proposed a first merging between MEC and NFV MANO whereas ETSI followed up in 2018, with the goal of building a MEC system on top of the NFV MANO framework and connecting together the MEC entities with the NFV MANO entities [4]. Note that ETSI takes a few assumptions that enable the integration of MEC and NFV [4].

- The MEC platform is deployed as a group of Virtual Network Functions, according to [22].
- MEC applications are independent VNFs with respect to the NFV MANO components. This allows the delegation of orchestration and life cycle management procedures to the NFV Orchestrator (NFVO) and the VNF Manager (VNFM) [22].
- The Virtualization Infrastructure became the Network Function Virtualization Infrastructure block (NFVI), managed by the VIM.

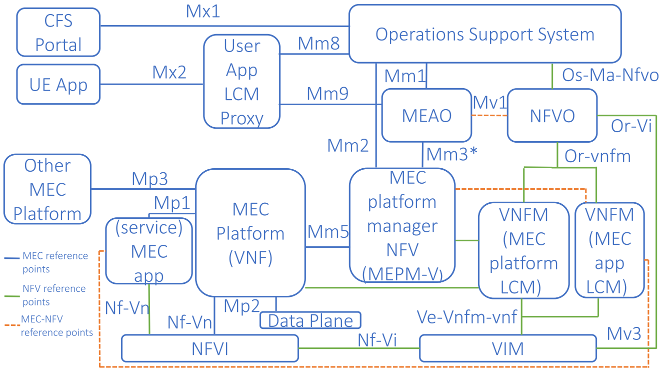

Fig. 5. MEC with NFV MANO architecture.

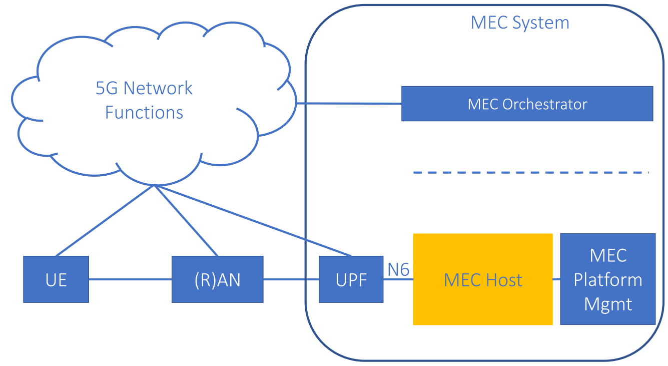

Fig. 6. MEC with 5G.

Fig. 5 shows the new MEC architecture leveraging on ETSI NFV MANO. It is possible to see that the MEC platform manager is now the MEC Platform-NFV (MEPM-V), delegating the applications life cycle management part to a dedicated Virtual Network Function Manager (one for every MEC host or even for every MEC application). The MEO is now the Mobile Edge Application Orchestrator (MEAO) and it is directly connected to the NFVO, via the *Mv1* link, which is used for the discovery and management of NFV network services such as a number of VNFs connected and orchestrated. The MEAO is also connected to the MEC platform manager through the *Mm3** reference point, which is based on *Mm3*.

As highlighted in [4], new reference points are introduced: *Mv1*, *Mv2*, *Mv3*. The first one connects MEAO and NFVO for the management of network services, while the second one links the MEC platform with the VNF manager of the application life cycle, giving the possibility to allow life cycle management related notifications to be exchanged between these entities. Finally, the third one allows for the exchanging of messages related to initial deployment-specific configuration of MEC application life cycle management.

Moreover, since the MEC architecture is deployed on top of an NFV scenario, the data plane is realized in two different ways: as VNFs connected to the MEC applications through the *Mp2* reference point, or by reusing the Service Function Chaining (SFC) functionality provided by the NFVI for traffic routing, without the need for a dedicated component and the *Mp2* reference point.

C. MEC and 5G

Together with SDN and NFV, MEC is a key pillar of 5G since early discussions [3]. In fact 5G networks require tight constraints on bandwidth and latency, achievable only by moving computing resources from the network core to the edge [24]. At the same time, operators are transforming themselves into vendors of versatile service platforms, so that the MEC concept becomes desirable for them [3].

The concept of MEC had been already partially standardized in a 4G context, when 5G requirements and the actual design were still in a primordial phase. However, the deployment of MEC in 5G is different from the one for 4G. MEC was an add-on for 4G, which was already deployed, when ETSI firstly introduced the MEC. Instead, 5G has been holistically designed with the MEC [25]. In particular, ETSI standardization efforts are built on top of the 3GPP specifications for 5G systems (such as 3GPP TS 23.501 [26]), allowing therefore the mapping of MEC blocks onto Application Functions (AFs) of 5G.[4] This offers the possibility to use services and information of 5G 3GPP network functions in the MEC. Furthermore, new functionalities have been defined with the goal of providing flexible support for several MEC deployments, taking into account MEC support for user mobility.

Fig. 6 shows the integration of MEC in 5G. Since this survey focuses mainly on MEC, the figure only shows 5G network functions actually needed for MEC deployment. The User Plane Function (UPF) is the most important one. UPF is a distributed and configurable data plane (seen from the MEC perspective), in charge of routing user plan traffic to the appropriate Data Network (DN). Its deployment is coupled with the one of the MEC hosts, which is either located in the same DN, to achieve low latency and high throughput at the edge, or reachable through the N6 reference point, which could be external to the 5G system, thanks to the deployment flexibility given by the UPF. Focusing on the MEC control side, the MEC Orchestrator can interact with the 3GPP Network Exposure Function (NEF) or with the target 5G network function.[5] At the MEC *host level*, the MEC platform can interact with the 5G network functions. MEC hosts will be deployed either at the edge or more inside the mobile network, even at the core of the network. It is responsibility of UPF to steer the traffic towards the targeted MEC applications. Moreover, in [26], 3GPP presents the most important enablers for edge computing, which are fundamentals for a correct MEC deployment in 5G networks [25]. These enablers are:

- *Local Routing and Traffic Steering:* the 5G core network architecture allows to route and steer traffic inside the local data network. AFs can also define specific traffic rules.

[4] Application Functions are logical elements of the 5G architecture defined by 3GPP. They provide session-related information, used to enable the interaction between control-plane Network Functions.

[5] Other 5G network functions are Network Resource Function (NRF) and Network Slice Selection Function (NSSF). For more details, please refer to [25].

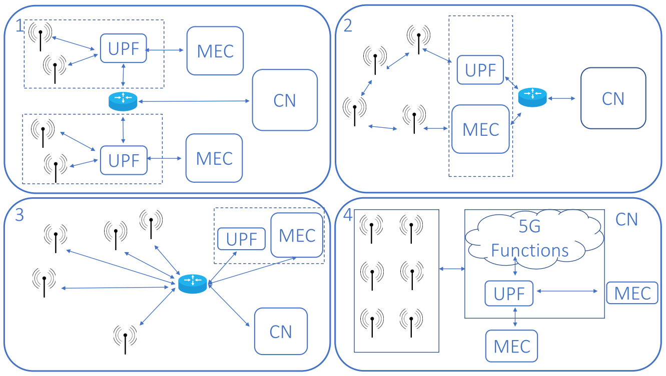

Fig. 7. MEC deployment scenario in the 5G context.

- *User plane Reselection and Selection:* AFs can define UPF traffic routing and (re)selection. This depends on the UPF deployment scenario and on the configuration of MEC services.
- *(Support of) Local Area Data Network (LADN):* this is enabled thanks to the UPF location flexibility, allowing to deploy MEC hosts between UPFs and a data network.
- *Session and Service Continuity (SSC):* It allows MEC to have full support for user and application mobility.
- *Network Capability Exposure:* through the NCE, the MEC has indirect access to 5G network functions.
- *QoS and Charging:* this makes it possible to route traffic to a LADN according to the QoS required.

Moreover, ETSI has recently published recommendations for the MEC support network slicing [27]. According to recommendations, entities such as MEAO, MEC platforms and MEPM-V should be aware of slices. Therefore, ETSI proposes to expand the reference points between these entities in order to include information on network slices. This revealed to be a very powerful tool. Indeed, based on ETSI recommendations, and on the results presented in [27], Ksentini and Frangoudis [28] were able to design an ETSI MEC orchestration/management architecture for network slicing, compliant with both ETSI and 3GPP. However, ETSI recommendations for network slicing still have several shortcomings. In [29], the authors addressed those limitations and proposed two solutions for a multi-slice MEC support: a Slice Control Function (SCF) in order to deploy slice-aware MEC App allocation and a inter-slice communication channel to allow exchanging of data in the same MEC facility.

Fig. 7 shows four deployment possibilities of MEC in 5G networks [25]: *(i)* MEC and UPF collocated together with the gNB, *(ii)* MEC deployed with a transmission node, possibly with a local UPF, *(iii)* MEC and local UPF located together with a network aggregation point, and *(iv)* MEC collocated with Core Network functions, inside a data center. The options presented above show how MEC can be flexibly deployed in different locations from near the gNB to a remote data network, which means that, notwithstanding its name, the MEC does not necessarily run at the edge of the mobile network![6] The UPF is deployed and used to steer the traffic towards the targeted MEC applications and towards the network.

On the 3GPP side, there are several technical reports that explain how to deploy MEC in 5G networks. For instance, 3GPP SA2 TR 23.748 [30] provides suggestions for several edge computing architecture enhancements in the 5G core network (5GC). The key system enhancements consist in:

- methods to discover the application server IP address at the network edge;
- 5GC enhancements to support seamless migration of application servers;
- methods to provide local application servers with network and/or traffic information, in a small amount of time;
- support for traffic steering in a edge N6-LAN.

That document also provides deployment guidelines for use cases such as URLLC, CDN, V2X, AR/VR.

In SA6 TS 23.558 [31], 3GPP specifies the application layer architecture (based on previous 3GPP technical reports), procedures and information flows needed for a correct deployment of edge applications over 3GPP networks. Further, they provide a first high level example on how their application layer architecture would merge with ETSI MEC. Finally, in TR 23.758 [32], 3GPP specifically studies architecture requirements for authentication of clients and discovery of edge services, stating that the mapping of those entities and ETSI MEC is considered as future work.

Finally, MEC opens new possibilities for business models. While in the simplest case the MEC could follow the IaaS, PaaS, or SaaS business architectures like the cloud computing paradigm does, other possible business models in a MEC multi-domain architecture are still unclear. This is because several players such as Mobile Virtual Network Operators or local 5G operators will compete to share the same MEC resources. In this scenario, some issues arise like, e.g., business coordination between domains, the setting of Service Level Agreements (SLAs) and pricing schemes [33]. A possible promising solution seems to consists in enhancing the reference points, in order to enable cross-actor interactions [33], [34]. Anyway, almost no papers in literature has yet addressed these issues, especially in industrial vertical domains (see Section IV).

D. Other Initiatives

ETSI has also standardized MEC features beyond the NFV MANO framework and 5G. At the same time the MEC concept is being explored by other organizations and in research projects.

Other ETSI contributions: For instance, ETSI has studied the pairing between MEC and Cloud RAN [35], which proposes the deployment of small data centers near the RAN. The benefits of this co-deployment are the mitigation of CAPEX and OPEX costs.

Another ETSI white paper [36] deals with MEC deployment in an enterprise environment, whose goals is to bring an

[6]Running MEC hosts far from the edge will be useful in scenarios in which compute power requirements are tighter than latency ones.

additional level of security, since data analytics will be performed locally, and a full support to different access networks. ETSI presented also several use cases, such as smart enterprise buildings, augmented reality conferencing and local data analysis. Note also that several MEC proofs-of-concept have been developed[7] according to the ETSI framework proposed in [20]. Finally, in 2018 ETSI published a white paper on the MEC support for V2X uses cases [37]. They provided several use cases examples for four macro groups (safety, convenience, advanced driving assistance, vulnerable road user), evaluating existing gaps and requirements for new MEC functions and features.

Standard Development Organizations (SDOs): Besides ETSI, there are other several SDOs that are working on the (edge) network ecosystem, in order to avoid disorganized works and redundant overlaps [38]. Some examples are, apart from ETSI, IEEE Edge Automation Platform (EAP),[8] 3GPP [39] and the O-RAN Alliance.[9] While EAP helps in giving high-level perspectives and insights, the work of 3GPP and O-RAN contains key constructs, relevant for a possible edge architecture, such as separation of control plane and user plane, access convergence, URLLC (from 3GPP) and intelligent *disaggregated software-based* RAN systems (from O-RAN [40]).

The O-RAN Alliance has been established in 2018 by several mobile network operators. Its mission is threefold: *i)* to develop new intelligent AI-enabled RAN, *ii)* to maximise the use of common-off-the-shelf hardware and *iii)* to specify new open source APIs and interfaces. These goals can potentially lead to several benefits such as decreased CAPEX/OPEX costs, improved network performance and efficiency and a greater ability to import new network capabilities. The O-RAN architecture is built upon the ETSI NFV specifications and can be also leveraged by ETSI MEC.

Indeed, a possible merging (or collaboration) between both architectures would enable, for instance, more context-aware MEC applications in V2X use cases or allow true network slicing at the edge (with improved QoS and/or QoE) [40], [41].

Other initiatives: Besides SDOs, there are several attempts to steering the MEC design, especially through Open Source projects. In 2016, Juniper published a white paper [42] discussing the benefits of edge computing and its use cases. The GSMA organization has focused on the impact of MEC on the mobile subscriber's experience.[10] The Edge Computing Project Group[11] has proposed the implementation of MEC-like applications and services at the edge of the network, with the goal of building a platform for 5G and IoT services. The 5G-MiEdge project [43] discussed the integration of MEC and mmWave technologies, with a use case based on the upcoming 2020 Olympics that will be held in Tokyo, Japan. Furthermore, the Industrial Internet Consortium, called in the past OpenFog Consortium,[12] is closely related to MEC, since it promotes the dissemination of Fog Computing with a special focus on the Industrial Internet of Things (IIoT). EdgeX Foundry[13] is a vendor-neutral open source platform that is placed at the edge of the network with the goal of building a MEC-like framework for IIoT. The EdgeX project is under the Linux Foundation Edge (LF Edge) project, together with Akraino Edge Stack,[14] Edge Virtualization Engine (Project EVE),[15] Open Glossary of Edge Computing,[16] Home Edge,[17] and Open Network Automation Platform. (ONAP)[18] The Open Edge Computing Initiative[19] is instead a project that aims to drive the business opportunities and technology rising from edge computing. It deployed the *Living Edge Lab* testbed for edge computing trials, located in Pittsburgh, Pennsylvania. Very closely related to MEC, the CORD project promises to combine NFV, SDN, and the elasticity of commodity clouds to bring data center economics and cloud agility to the telco central office.[20] MobileedgeX, founded by Deutsche Telekom, is creating a marketplace of edge resources and services that will connect developers.[21]

ETSI MEC extensions: Several papers propose MEC architectures and extensions. Arora *et al.* [44] propose a new MEC architecture for the Radio Network Information Service (RNIS), based on OpenAirInterface and fully compliant with the new ETSI MEC in NFV standard [4]. This service, present in the MEC platform, allows edge applications to know RAN conditions, so as to be able to modify their behavior and match the network conditions [45]. They create two different message-brokers of the RNIS, one with RabbitMQ and the other with Kafka, with the first one being superior in terms of lightweight CPU utilization.

Zanzi *et al.* [46] focus on the introduction of a MEC *Broker* on top of the ETSI MEC architecture, between the OSS block and the tenant (i.e., the UE). The MEC broker enables tenants to access management options such as life cycle management and application administration privileges. In addition, they propose an orchestration solution called M^{2}EC (from multi-tenancy MEC), which allows for minimizing overall resource utilization. Castellano *et al.* [34] propose a split MEC architecture, in contrast with the current monolithic ETSI MEC architecture we have described in Section II-A. They argue that standards are not helping the MEC deployment in real scenarios and, at the same time, companies are looking at MEC as an opportunity to save money or generating revenues. Therefore, they propose to further separate the ETSI MEC architecture in Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) levels, in order to help the MEC deployment. Huang *et al.* [47] present in details an SDN-based MEC framework, compliant with both ETSI MEC and 3GPP architectures. According to the authors,

[7]https://mecwiki.etsi.org/index.php?title=OngoingPoCs
[8]https://futurenetworks.ieee.org/
[9]https://www.o-ran.org/
[10]https://www.gsma.com/
[11]https://telecominfraproject.com/
[12]https://www.openfogconsortium.org/
[13]https://www.openedgecomputing.org/
[14]https://www.lfedge.org/projects/akraino/
[15]https://www.lfedge.org/projects/eve/
[16] https://www.lfedge.org/projects/openglossary
[17]https://www.lfedge.org/projects/homeedge/
[18]https://www.onap.org/
[19] https://www.openedgecomputing.org/
[20]https://opencord.org/
[21]https://mobiledgex.com/

it provides the required data-plane flexibility and programmability, improving overall latency. Finally, Taleb *et al.* [48] propose the concept of a Content Delivery Network (CDN) slice, which is a CDN service instance created upon a content provider's request. They base their proposal on the latest versions of MEC, NFV and on proposal produced in the frame of 5G standardization efforts. In their work they tackle QoE-driven cloud resource allocation and elastic resource management.

III. MEC Flexible Provisioning

MEC provisioning is an important feature because, thanks also the degree of flexibility provided by NFV, it will help the MEC paradigm to set itself with a primary role in the deployment of future Internet architectures. Efficient provisioning is achieved by means of both careful MEC resource deployment and the capacity to follow the user mobility. Therefore, this section answers the question *Where should MEC resources be deployed?*. To answer, we cover two important MEC aspects: (*i*) (flexible) MEC resources deployment and (*ii*) agile migration of MEC resources. Indeed, it is of crucial importance for some applications that devices can reach MEC resources in few milliseconds or less (i.e., with extremely low latency) and that edge resources are fairly assigned to services [49]. Moreover, the MEC must also support user mobility, which requires rapid service provisioning and fast migration of applications, VNFs and MEC services. These features enable innovative solutions for a well-known MEC research problem, i.e., for computation offloading.

Section Organization: In what follows, we will overview the most relevant papers regarding MEC resources deployment in Section III-A, agile resource migration in Section III-B and afterwards we will focus on the most recent papers on MEC with computation offloading III-C. Finally, we identify lessons learned and future research opportunities in Section III-D. Table III contains a summary of reviewed papers, divided by use cases, analytical tools and evaluation methods.

A. Flexible MEC Resources Deployment

Location deployment: A fundamental problem studied in the literature concerns where to physically deploy MEC resources. Some examples can be found in a recent Intel white paper [50] and several papers in the literature address this topic.

In [51], Pérez *et al.* highlight that in future deployments, mobile network operators will have to decide how many MEC *points of presence* are needed, considering also the presence of gNBs. Therefore, they created a model, based on inhomogeneous Poisson point processes, which studies the MEC deployment with simulations based on a real topology. Since MEC deployments are constrained by the cell tower presence, Syamkumar *et al.* [52] analyze a 4M dataset of antennas located in the U.S. in order to evaluate the MEC deployment in a real case scenario, showing in which areas new network infrastructures are needed. Similarly, in [53], the authors study how to allocate MEC resources as a function of service demand. They propose a graph-based algorithm to provide a partition of MEC clusters, which takes into account the capacity of MEC servers. The authors evaluated it with a mobile communications data set, containing real world spatio-temporal human dynamics. Furthermore, in [54], the authors study how the mobility of citizens in a city should also affect the optimal placement of MECs. Kherraf *et al.* [55] formulate the problem of MEC resource provisioning and workload assignment for IoT services (RPWA) with a mixed integer programming formulation. Given its complexity they decouple the problem into two sub-problems: (*i*) delay aware load assignment and *ii)* mobile edge servers dimensioning. Through numerical simulations, they show that their scheme achieves a higher admission rate (from 1% to 44%) compared to the solution proposed in [56].

Filippou *et al.* [57] provide another way to deploy efficiently MEC resources. Indeed, the authors focus on the control plane, studying the latency of packet transfer and processing inside an NFV environment. To minimize the latency, they design proximity zones around MEC platforms hosting MEC application instances, showing how these zones could help for a flexible and latency-aware use of the MEC platform. Castellano *et al.* [58] propose a distributed algorithm to coordinate the resource allocation in edge computing scenarios. They take into account the optimal resource assignment and evaluate its feasibility with a prototype implementation that follows a Pareto-optimal resource assignment.

VNF placement at the edge: MEC resource placement can be made flexible thanks to the use of VNFs on top of virtualized infrastructure, using virtual machines and containers. Hence, new scenarios are now available to be explored together with the MEC paradigm: the VNF placement and resource migration (which this survey will overview in the following sub-section). Depending on the type of service, different constraints (e.g., low latency, high compute power and/or a fixed dedicated uplink/downlink bandwidth) are present and the MEC Orchestrator should be able to decide quickly where to place the VNFs, e.g., near the core of the network or at the edge.

In [59], the authors propose a data-driven VNF placement strategy with ONAP across distributed data centers, hence in a MEC scenario. Through simulations, they compare their solution against other ones proposed with an Openstack-based approach, showing that their strategy is better in terms of overhead and data center utilization. Salsano *et al.* [60] propose an architecture for the dynamic deployment of VNFs leveraging on the MEC. According to the principles designed in the SuperFluidity project, they decomposed the network functions needed for MEC as software *reusable functional blocks* (RFB), which hence allows for flexibility in the architecture. The proposal has been validated by studying a video streaming service use case. In [61], the authors build a VNF placement strategy on top of ETSI standards for MEC and NFV MANO. They propose a genetic algorithm, considering as constraints access latency and service availability. Through numerical results, they show the feasibility of their algorithm, reaching near-optimal performance.

Poularakis *et al.* [62], focus on joint service placement and request routing problem in a MEC multi-cell scenario with multiple constraints, aiming to minimize the load of the

TABLE III
LIST OF PAPERS SURVEYED IN SECTION III

MEC Provisioning	References	Use Case	Analytical tools	Evaluation	Most relevant lessons learned
Computation Offloading	[89], [90], [91] [92], [95], [96] [97], [98], [99] [101], [102], [103]	- IoT - Privacy preserving - Inter task dependency - Energy efficiency - Parallel computing - Autonomous devices - Caching - M2M - Wireless Energy Transfer (WET)	- Mixed Integer Program (MIP) - Logic Based Benders Decomposition - Lyapunov optimization - Gibbs Sampling algorithm - Markov Decision Process - Deep Learning - Non-convex MIP - Dinkelbach's method - Game theory	- Numerical simulations	- Adding privacy constraint does not affect (very much) performance ($< 5\%$) - Deep Learning algorithms can save energy up to 87% compared to baselines
Flexible MEC resources deployment	[51], [52], [53] [54], [55], [56] [58], [59]	Location deployment	- Model with inhomogeneous Poisson point processes - Voronoi cell-based analysis - Graph-based algorithm - MIP	- Simulations with real scenarios - Datasets - Numerical simulations - Prototype implementation	- Considering the New Radio profiles of 5G, FDD 120 kHz is the one that minimises the number of MEC stations deployment - Up to 50% of traffic can be absorbed by MEC servers
	[60], [61], [62] [63], [66], [67] [68]	VNF placement at the edge	-Architecture design - Genetic Algorithm - Randomized Rounding algorithm - Polynomial-time algorithm	- Numerical simulations - Prototype implementation - Trace-driven simulations	- Randomized Rounding algorithms could be a viable solutions for VNF placement at the edge, with performance close to the optimal
	[69], [70], [71]	Systems deployment	- Framework design - MEC platform deployment	- Systems-level evaluation - Open Air Interface prototype	- MEC over FiWi could prolong devices battery up to 11.30 h - MEC can reduce latency up to 60% compared to a cloud datacenter
Agile Migration of MEC-VNF-based resources	[72], [73], [74] [76], [77], [78] [79]	Mobility support	- MDP - Integer Linear Programming	- Numerical simulations - Testbed experiments	- Follow me Cloud with MEC can reduce the iterative migration time up to 61% compared to existing solutions - Increasing the number of service replicas reduces the probability of user reactive migration (from 21% to 26.5%)
	[80], [81], [82] [83], [84], [85] [87], [75], [88]	Migration with containers	- MPD - Optimal stopping theory	- Testbed experiments - Data traces simulations - Prototype implementation - Numerical simulations	- Containers reduce up to 56% the handoff time - A dynamic placement scheduler reduces VNFs migrations up to 94.8% compared to baselines schedulers - Containers achieve from 2x up to 8x faster migration time compared to VMs migration

centralized cloud. They propose a custom *randomized rounding* algorithm, showing that, in terms of cloud load, they can achieve a 25% better performance with respect to the greedy solution proposed in [63]. Similarly, the authors of [64] explain how to design an edge computing framework, which also includes a service orchestration algorithm. The latter allows to move and place services within 25 ms and it has the ability to scale and support services instantiated on a per-user basis. In [65], the authors propose a two-scale framework that jointly optimizes service placement and scheduling of requests under storage, communication, computation, and budget constraints, proving that the problem is NP-hard. Furthermore, they develop a service placement polynomial-time algorithm which reaches performance close to the optimal solution (up to 90%).

Moreover, some papers deal with an edge-cloud architecture. Yang *et al.* [66], for instance, study the problem of service chaining with VNFs in a mixed edge-cloud scenario. They minimize the maximum link load ratio under delay constraints. Finally, the authors in [67] study the optimal provisioning of edge services with both shareable and non-shareable resources via joint service placement and request scheduling. They show that the problem is NP-hard and propose several heuristics which are then evaluated via data-driven simulations.

Systems deployment: Some works dig more on system implementation. Rimal *et al.* [68] propose a MEC deployment over Fi-Wi, which is a combination of mmWaves and optical fibers that allows to achieve ultra-high speed. The authors discuss the possible benefits of the framework, such as prolonging the discharge of edge device batteries, with a capacity of just 1000 mAh, up to 11.5 hours, depending on the offloaded traffic load. In [69], the authors provide a MEC platform deployment solution for 4G LTE networks using a middlebox, for which they have designed a prototype based on the OpenAirInterface (OAI) cellular platform. Other works propose the integration with different technologies. For instance, in [70], van Kempen *et al.* provide the design of the so-called MEC-ConPaas platform, a mobile-edge cloud platform that aims to support future research on edge cloud applications, leveraging on Raspberry Pi devices. Their experiments show that it is possible to support real cloud applications with extremely simple edge devices.

B. Agile Migration of VNF-Based MEC Resources

In addition, due to user mobility, it is of primary importance to establish a connection between the end user and MEC resources, and maintain it throughout all the necessary stages, with the services that should be able to migrate quickly depending on user movements.

Mobility support (Follow me Cloud and Service Replication): The relation between MEC and mobility of users and the dynamics of their demands is also object of investigation. Several works addressed the performance and optimization strategies for migration in a MEC scenario, to keep performance levels high and use resources efficiently.

One particular paradigm developed for user mobility support is the so called Follow me Cloud, which has been proposed in [71]. Follow me Cloud uses an approach similar to Information Centric Networks (ICN): it proposes the replacement of the IP addressing for a service/data identification. This allows for a continue connection between mobile user and service, even when service migration occurs. After the initial paper in 2013, Follow me Cloud has been further extended. In [72], the authors compare the Follow me Cloud paradigm with other two testbeds based on locator/identifier separation protocol (LISP) and SDN, showing the potential of their paradigm and its feasibility for real-world deployment.

In [73], Addad *et al.* merge the Follow Me Cloud concept with the MEC paradigm, in order to provide lightweight live migration at the edge, based on container technologies. They evaluate their proposal with a real testbed. According to their results, using Follow me Cloud with MEC would decrease iterative migration time by 50% compared to the baseline solution proposed in [74].

Finally, in [75], the authors focus on the vehicular networking case and develop a new architecture named Follow me edge-Cloud (FMeC). Leveraging on the strict requirement of the automotive vertical, they created an FMeC architecture based on MEC and SDN/OpenFlow principles, and validated their new concept through theoretical analysis and simulation experiments. Instead, Farris *et al.* [76] study the proactive service replication problem, to reduce the overall migration time and guarantee good QoE. They leverage the prediction of user mobility patterns and the overall synchronization of states of service replicas. At the same time, this technique collides with limited edge resources. Therefore, the authors formulate two different optimization problems: one minimizes QoE degradation during handover, while the other minimizes the cost of service replicas. Through simulations, they show that increasing the number of service replicas would reduce the probability user reactive migration by up to 26.5%.

Similarly, the authors of [77] deal with the fast relocation problem of services due to user mobility, investigating container-based virtualization techniques. In their work, they support the use of mobility in a MEC infrastructure by designing a framework with three different modules (*Service Manager* for monitoring applications, *Edge Manager* for containers placement and *Edge Orchestrator*, which manages the overall framework) to guarantee fast response time and exploiting service replication. They show the benefits with respect to classic migration procedures. They further state that this framework may also be integrated into the ETSI MEC architecture. Finally, Sangaiah *et al.* [78] propose to leverage machine learning techniques on MEC nodes for preserving position confidentiality of roaming users, arguing that MEC servers would help for maintain both a low latency service and position confidentiality.

Migration with containers: Resource migration mainly deals with VNFs, e.g., migration of the virtual machines and containers that run the VNFs, across different hosts. Normally, in a centralized data center most of the virtualized resources and migrations are for virtual machines. However, placing resources at the edge of the network leads to the deployment of small data centers in which it is not possible to execute the same virtualization technologies of typical large data centers [79]. Therefore, services would be better deployed using containers, which represent a lightweight solution for deployment and migration. Therefore, many studies focused on container migration. The authors of [80] evaluate Docker, which is the most commonly adopted and powerful container technology as of today, at least in the scenario of edge computing. They base their evaluation on four different aspects: deployment and termination, resource and service management, fault tolerance, and caching. They show that Docker is a valid candidate platform for edge computing. Furthermore, Avino *et al.* [81] state that a key beneficial feature of MEC would be the ability to ensure server portability with low overhead. They show that this can be achieved using Docker. To prove this, they quantify Docker CPU utilization in two use cases in an experimental setup: online gaming and video streaming. In both cases the Docker overhead was quite small, even though for the online gaming case the overhead slightly increases with the number of supported servers. Wang *et al.* [82] state that in a MEC scenario, the migration of resources is difficult to perform since the environment is very dynamic and volatile. Hence, they propose a Markov decision process to deal with this uncertain scenario, validating their model by means of mobility traces for San Francisco taxis. Recently, Doan *et al.* [83] have proposed a measurement framework in order to study the existing data center migration approaches in a MEC scenario. They show that these approaches are unfeasible due to the high migration time, causing therefore substantial service degradation.

The papers mentioned above do not consider stateful migration. With stateful migration, the service is migrated and resumed in the exact state in which it was before migration, without losing connection with the users. In [84], the authors' goal is to achieve a seamless live migration, with focus on reducing the file transfer size during the migration procedure. They study Docker layered storage and propose to share common storage layers across Docker hosts in order to reduce file transfer size. They propose and evaluate a prototype, which shows interesting performance improvements (up to 56% reduction of hand-off time with respect to reference approaches defined in [85]). In [86], the authors argue that containers would be fundamental for meeting low latency requirements. They study state of the art of the migration techniques with Docker and with virtual machines using KVM. Moreover, they propose an application level live migration protocol that eliminates common drawbacks like the lack of hardware abstraction at the host. The work of Machen *et al.* [74] proposes a 3-layer framework for supporting stateful live service migration encapsulated in containers in a MEC scenario, with the goal to ease the implementation with popular container and virtual machine technologies. They validate their solution with small scale experimental results, showing that containers can achieve from 2x up to 8x (depending on the scenario evaluated) faster migration times compared to VMs migration.

Finally, Cziva *et al.* [87] propose a more general framework on VNF migration. They focus on a dynamic placement of VNFs at the edge of the network and especially on the dynamic re-schedule of VNF placement. Their approach leverages optimal stopping theory. They run simulations based on a nation-wide backbone network with real world ISP latency and show that their solution incurs much less VNF migrations (up to 94.8%) then other existing migration schemes.

C. Computation Offloading

A well-known research problem coupled with MEC is the computation offloading problem. Indeed, thanks to the deployment of edge resources and their agile migration, it is possible to offload computation tasks from mobile users, with benefits for instance on the device battery life. Computation, according to [17] could be *fully* offloaded to the MEC, *partial* offloaded, or utterly processed at mobile device (*local execution*). Hence, this new paradigm raises new questions and challenges in the MEC resources deployment domain, such as the trade-off between minimizing device energy consumption while achieving acceptable execution delay due to offloading. Of course, the delay also depends on the MEC resource deployment [17]. Another problem is in identifying the edge server that should be selected for offloading.

In [88], the authors propose a delay-sensitive IoT services scenario, in which task offloading is jointly considered with (MEC) resource allocation and (task) scheduling. They formulate the mixed-integer problem "Dynamic Task Offloading and Scheduling (DTOS)". Due to its complexity, they decompose the problem using a technique called *logic based benders decomposition*, and perform several simulations in order to check the effectiveness of their proposed solution. Finally, with the same algorithm, they evaluate trends in different vertical industries, namely tactile Internet, tele-surgery, Factory, Automation, ITS and Smart Grid, with variable latency requirements.

In [89], the authors focus on the privacy aspect of offloading to a MEC server. They show that existent privacy-preserving techniques do not work well in this new edge scenario and so they create PEACE, a scheme that jointly considers privacy-preserving and cost-efficient task offloading. According to their experiments, adding the privacy constraint does not affect very much the overall performance ($\approx$ 5%). Yan *et al.* [90] study the inter-user task dependency in an MEC system. First, they focus on a 2-user MEC scenario (in which the input task of a user requires the output task of the other user). Their goal is to minimize both the energy consumption of users and tasks execution time through an optimal task offloading policy and resource allocation problem; the problem is further solved by means of a reduced complexity Gibbs sampling algorithm. Further, they extend the scenario to a general multi-user MEC, in which an input task of a user requires final task outputs from multiple users. They evaluate this extension with the same algorithm proposed for the single-user case and find that their solution performs well compared to other sub-optimal schemes. Meng *et al.* [91] aim to achieve a delay-optimal computation offloading policy for computation constrained MEC systems, taking into account also the future delay performance of the MEC system. In order to deal with this problem, they create a finite horizon Markov decision process (MDP) for two cases: single-user single-MEC server and multi-user multi-MEC server scenario. They manage to derive a closed-form multi-level water-filling computation offloading solution and show via simulation that it outperforms other schemes proposed in [92] and [93] by $\approx$ 4% in terms of average delay.

In [94], the authors' goal is to improve energy efficiency of a MEC system hosting both URLLC and delay-tolerant services. To solve this problem, they use a Deep Neural Network (DNN), trained with a so-called digital twin model (a virtual digital mode that merges data from the real network and fundamental rules from theoretical studies), showing the benefits of their DNN framework. Compared against baselines, it enables energy savings up to 87%. In [95], the authors state that virtualization on shared I/O resources, which could happen in an edge computing scenario, might lead to computation degrading (meaning that the speed of VMs sharing the same hardware might degrade due to interference). Therefore, they study the problem of joint radio-and-computation resource allocation (RCRA) in multiuser MEC systems in the presence of I/O interference, showing that their solution performs well against optimal algorithms ($\approx$ 4% of difference). In [96], Josilo *et al.* focus on the coordination problem of offloading to the MEC decisions of autonomous devices, such as vehicles, drones or manufacturing machines, with the goal of minimizing device energy consumption and task completion time through a game theoretical analysis. Wang *et al.* [97], portrait a joint optimization problem on the computation offloading and content caching strategies for wireless cellular networks with MEC. They propose an alternating direction method of multipliers (ADMM) algorithm, evaluating its effectiveness with different system parameters.

In [98], the authors focus on the integration between virtualized small cell networks (SCNs) with MEC. Their solution might help in reducing the energy consumption of UEs thanks to offloading procedures. However, complexity might explode. The authors formulate the problem as a mixed integer nonlinear program and then transform it into a biconvex problem. Through simulations, they compare it against the optimal and an algorithm proposed in [99]. Their solution achieves better performance, with a gain of about 20% against [99], while nearly reaching optimal performance.

The authors of [100] state that nowadays machine-to-machine (M2M) communications attract ever growing attention. Differently from other communications networks, M2M uses high-frequency small packet size, therefore needing a special optimization of both energy consumption and computation. Therefore, the authors introduce a MEC architecture for virtualized cellular networks with M2M communications, to decrease energy consumption and optimize the computing resource allocation. They create an observable MDP to minimize the system cost. Mao *et al.* [101] propose to use Wireless Energy Transfer (WEF) to prolong device battery life. However, it is hard, for the MEC system, to jointly schedule radio and computational resources as well as energy utilization maintaining at the same time the overall performance requirements. Hence, they study energy efficiency and delay in a multi-user wireless powered MEC system with multiple access schemes. They design a low-complexity online algorithm based on Lyapunov optimization theory, allowing to transform their problem into a series of deterministic optimization problems. Through theoretical analysis they show that their algorithm allows to trade off energy efficiency for delay.

Sardellitti *et al.* [102] formulate the computation offloading problem, from the mobile users to the cloud server, in a multi-cell mobile edge computing scenario. They define it as the joint optimization of radio and computational resources with the goal of minimizing multi-user's energy consumption under latency constraints. They find that in the MEC scenario, offloading becomes more convenient with high computational loads.

D. Summary, Lessons Learned and Future Research

We now summarize lessons learned from the overviewed papers, highlight potential improvements to existing solutions. We also identify possible future research directions.

MEC deployment: One of the most important novel aspects of MEC is its proximity to the UE. This leads to new unexplored scenarios and gives the possibility to enhance different features, such as computation offloading. However, MEC provisioning is challenging. Indeed, network providers should carefully consider both the QoS required by services (e.g., the ones using URLLC) and the cost of deploying and maintaining a new edge infrastructure. In [50], in order to achieve this tradeoff, the authors propose to expand the existing infrastructure, i.e., network provider's towers and offices. Otherwise, new deployment possibilities lay inside the *last mile* network, thus helping with the development of the *smart city* paradigm. Hence, new sites could be stadiums, private/public building, enterprises or homes. Moreover, the NFV paradigm introduces a degree of flexibility. For instance, it will possible to create a disaggregated MEC architecture, where the MEC orchestrator is placed in a more centralized node and MEC hosts are instead more decentralized, nearer to users. At the same time, NFV allows MEC to support user mobility, with the migration of virtual resources across the edge infrastructure.

From the papers surveyed, the key **lessons learned** are:

- Computation offloading is one of the most studied paradigms within MEC and, in general, edge computing. This survey overviewed the most recent papers in this area. Most of the papers focus on minimizing device energy consumption playing with offloading tasks, resource allocation or delays [88], [90], [91], [94], [95], [96], [98], [100], [101], and [102]. Among all results, we would like to mention the performance of deep learning which, according to [94], is able to save devices energy up to 87%. Moreover, the authors of [89] consider also privacy issues, showing that the privacy constraint does not affect performance very much (i.e., within $\approx$ 5%). However, it would be interesting to see more works that consider new scenarios such as user mobility within different MEC hosts and 5G features such as network slicing [27]. Applications scenarios, e.g., cloud gaming, are not fully explored. Similarly there is a need to study VNF (applications) sharing (like in [103]) or the revenues/economical costs of offloading decisions, possibly also leveraging artificial intelligence tools. For interested readers, we mention the survey of Mach and Becvar [17] on MEC with computation offloading.
- Several papers focus on the MEC location deployment. While [53] confirms the benefit stemming from the presence of MEC, other authors provided useful insights on possible MEC deployments according to 5G constraints [43], [51], [52]. They consider smart cities, as well as industrial and rural scenarios. The authors of [54] study the MEC deployment in a smart city, considering pedestrian mobility. The authors of [57] enhance MEC host deployment by designing proximity zones around MEC platforms, helping them to become more latency-aware. Finally, the authors of [58] and [55] provide a more theoretical approach. Reference [58] considers a decentralized orchestration while [55] show that in terms of admission rate their scheme reaches higher performance (from 1% to 44%) compared to [56].
- Afterwards, different papers focused on VNF placement at the edge. Several approaches have been proposed, ranging from different frameworks [59], [60], [64] to theoretical works [62], [65], [66], [67].
- Finally, some systems-related works have been highlighted [68], [69], [70],giving several interesting insights.

For instance, MEC reduces latency up to 60% compared to a cloud datacenter.

Concerning possible future work on MEC deployment, we mention the following points:

- It would be interesting to investigate more location deployments and VNF placement resources at the edge, in real-case system-oriented scenarios, leveraging new possibilities given by new network protocols, standardized interfaces, new technologies such as UAV ([104], [105]), techniques such as ML, but also keeping in mind issues such as network scalability and constraints such as QoS, QoE, CAPEX and OPEX.
- Future works on MEC deployments should also consider standardization efforts made by SDOs such as ETSI MEC and O-RAN, also pointing out possible shortcomings and filling these missing gaps (some examples are [28], [44], and [106]).
- Furthermore, researchers should exploit new scenarios as the ones proposed at the end of this section, under *other research challenges*.

MEC migration: Regarding the agile migration of VNF-based MEC resources, two main paths have been evaluated:

- **Mobility support** given by the stateful Follow me Cloud paradigm [71], [72], which is shown to work better than solutions based on LISP and SDN. The authors of [73] and [75] have merged that paradigm with vehicular networks and MEC, whereas [76] and [77] propose a proactive service replication in order to reduce migration time. They showed that increasing the number of service replicas would reduce the probability of user reactive migration (from 21% to 26.5%). Instead [78] proposes to leverage both MEC and machine learning for maintaining services position confidentiality.
- **Migration** of VNFs and especially containers, since the latter are more lightweight with respect to VMs (an important feature in a scarce-resource edge infrastructure). Indeed, according to [74], containers achieve from 2x up to 8x (depending on the scenario evaluated) faster migration times compared to VMs migration. This survey analyzed preliminary stateless migration papers ([80], [81], [83]) and more recent works, focusing on stateful (live) migrations ([74], [84], [86]). We have also addressed some more theoretical works ([82], [87]).

Future works on migration should consider the following points:

- There is a need for evaluating the performance of new networks protocols such as segment routing v6 [107] for the migration and connection of MEC resources, thanks to their ability to support Service Function Chaining.
- While VM migration has been deemed too much heavy and slow for an edge infrastructure, more work is needed for understanding stateful lightweight migration, also considering new paradigms such as serverless computing [108], which seems the most promising feature to guarantee smooth QoE.
- New works should also consider new scenarios mentioned in the following paragraph.

Other research challenges: Finally, some possible **open research challenges** can be identified:

- **Privacy** and **security** are still open challenges for MEC [7]. In the design of the new generation of network infrastructure, privacy, the protection of data in general and security are becoming new important constraints to take into consideration. So far, the literature provides only a limited overview ([89], [109], [110]), while many subjects (authentication between edge/core, proper encryption, how to provide access only to secure devices, etc.) remain unexplored. Moreover, security attacks can happen also during VM migrations, in compromised VNFs (which might be migrated and accessed in another location with less security policies) but also with physical hardware (power cutout or NFV state Manipulation Attack [111]). Researchers should take into account all these threats when considering resources migration or deployment. For interested readers, the survey of Khan *et al.* [111] outlines several security and privacy threats in 5G and NFV systems, commenting on potential solutions.
- The use of **artificial intelligence** has been so far very limited in MEC-related works, while it promises dramatic improvements in terms of efficiency and cost reduction, especially for use cases involving complex systems and cyber-physical systems. Exploiting new machine learning techniques such as Edge ML [112], federated learning [113] and distributed learning [114] would help in many MEC areas, such as *smart* deployment and migration of MEC resources, edge big data analytics and/or caching at the edge.
- While many papers try to minimize the energy consumption at the user side, it is still unclear how to minimize the energy consumption at MEC side, exploiting therefore the **green MEC** paradigm [15]. Already nowadays, data centers are one of the most energy consuming infrastructures, and the deployment of new resources at the network edge will surely increase the energy consumption, together with capital expenditures of network providers. Exploiting hence green energy at the edge (wind, photo-voltaic, etc.) represents a possible solution to overcome these issues [115] [116].
- Only few works like [117] addressed **revenues** and **monetization** topics. With the possibility to deploy new applications and services at the edge of the network, and thanks to 5G features such as network slicing, that allows for the sharing of network resources between several tenants, new interesting problems arise, such as the sharing of limited edge resources between several paying tenants or the admission control of requests based on revenue generation.
- Most of the works are theoretical and only few works are **systems-oriented** ([68], [69], [70]). A more system-oriented literature, considering hence standards or novel network protocols/infrastructures, would help in understanding the suitability of MEC in real-case scenario or in the wild. For instance, it would be interesting to evaluate the integration of MEC with several open source projects

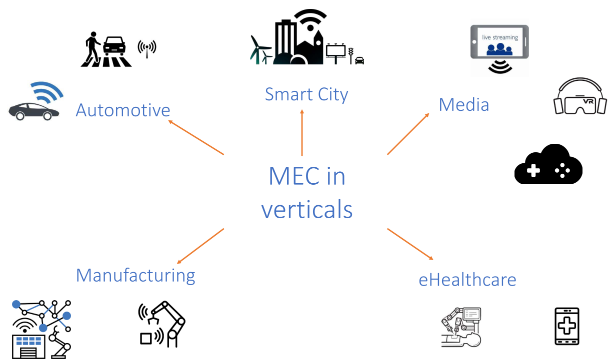

Fig. 8. MEC in industrial verticals.

and new Internet architectures such as hybrid ICN (hICN), Recursive Inter Network Architecture (RINA) and programmable-networks tools such as Open Flow or Net-FPGA [118].

- Finally, while 5G is slowly becoming a reality, preliminary works on 6G started to appear. Thus, even though we are still in a very early discussion phase, it would be interesting to study the role of MEC in the next generation of cellular networks.

IV. MEC in Vertical Industries

With network slicing, future networks will be able to serve different verticals at the same time. Nevertheless, verticals require different constraints, which can be handled through the MEC. For instance, the MEC will guarantee low latency computing resources with a high degree of flexibility offered to verticals and network providers. Here, we explore the impact of MEC on the most relevant vertical industries defined by 5G-PPP[22]: automotive, smart city, media, eHealthcare and manufacture, while in Section IV-F the survey focuses on lessons learned and future research challenges. Table IV shows a summary of surveyed papers focusing on vertical industries.

A. Automotive

From early 1990s, Intelligent Transportation Systems (ITS) have been studied in order to exploit communications between vehicles and infrastructures, with the goal of improving safety and efficiency of transportation. In this context, IEEE developed a new communication protocol called IEEE 802.11p [119]. However, that standard presents several limitations such as a poor scalability and lacks on performance guarantees. One way to help meeting the tight requirements of automotive systems is to leverage cellular networks, e.g., by means of the C-V2X (Cellular-based vehicle-to-everything) communications paradigm, firstly proposed for LTE and now extended to 5G networks. Especially 5G should become one of the most important enablers for vehicle communications since, thanks to SDN, NFV and MEC technologies, it aims to achieve high reliability jointly with low latency (i.e., with URLLC-based slices) [120].

[22]https://5g-ppp.eu/verticals/

Several papers pointed out the benefits for the automotive industry from leveraging MEC systems. In [121], the authors explain the motivations behind using MEC in ITS, stating that IEEE 802.11p and pure cellular networks might not be sufficient to serve for the stringent requirements of the automotive industry. Instead, using the MEC can guarantee reliable low latency communications, seamless service delivery and highly localized computing resources, necessary in order to achieve effective C-V2X connections, as further confirmed by means of extensive simulations in [122]. The authors of [121] also outline possible research challenges such as resiliency, security and privacy, resource management and orchestration, and cooperative awareness among the others.

However, in literature most of the works focus on technical challenges such as enabling edge communications with different access technologies, computation offloading, ad hoc computing resources (vehicular clouds) or supporting driving paradigms such as platooning.

For instance, Hu *et al.* [123] propose a MEC framework for automotive systems, composed by different communications technologies (mmWave, IEEE 802.11p and licensed sub-6 GHz band), with the goal of supplying services and contents to vehicles. They show through simulations that the adoption of three different access technologies outperforms solutions with only mmWave or sub-6 GHz + mmWave access technologies in various scenarios, especially in the highly dense and low bandwidth ones.

Furthermore, from 2017, the 5G Automotive Association (5GAA) defined the concept of Cooperative Intelligent Transportation Systems (C-ITS), stating that edge computing and in particular MEC will be the enabling technology for V2X communications. In their white paper [124], 5GAA proposes to categorize the main use cases into four groups (as the ETSI MEC standard for V2X does [37]):

- ***Safety:*** This group studies how to avoid collisions between vehicles, for instance at an intersection.
- ***Convenience:*** this group provides time-saving services to manage data and the health of the vehicle (such as the delivery and management of automotive software updates).
- ***Advanced Driving Assistance:*** It includes cases such as traffic signal timing. improving traffic flow, Real-Time Situational Awareness, Cooperative Lane Change (CLC) of Automate Vehicles and High Definition Maps. According to [124], for its processing of a large amount of data with low latency and high reliability, this is the most challenging use case for MEC.
- ***Vulnerable Road User (VRU):*** Finally, this group studies communications between vehicles and pedestrians.

Safety and VRU: In literature several papers focus on safety issues or VRU discovery with MEC. For instance, Nguyen *et al.* [125] discuss a method to avoid a collision between pedestrians and vehicles, deploying a MEC server near a base station. This deployment would help smartphones to save energy, giving the possibility to offload the calculation of the collision detection algorithm (CDA) to the MEC server and therefore avoiding both the smartphone battery drainage and calculation latency issues. Through simulations, they show that this solution would improve phone energy efficiency. In [126], the authors propose an enhanced collision avoidance (eCA) mechanism, placed in a MEC server, based on both a collision avoidance algorithm (CAA), and a collision avoidance strategy (CAS). The first algorithm evaluates future vehicle trajectories through beacons while the second strategy decides which vehicles should slow down to avoid collisions. They perform simulations based on SUMO and NS-3, showing the benefits of their strategy by reaching almost 100% of avoided collision in all the scenarios evaluated. Malinverno *et al.* [127] extend the collision detection algorithm, showed in [128], in order to avoid collision between pedestrians and vehicles, leveraging on a MEC-based architecture. Through a detailed simulation scenario, they showed that with autonomous cars 100% of the collisions can be detected on time before the accident happens, while with human drivers, the number decrease by 14%. In [129], Avino *et al.* developed a MEC platform based on ETSI standards and OpenAirInterface, in order to support automotive systems with tight latency requirements such as safety services. In their simulations, they show that it is possible to obtain better performance in terms of end-to-end delay with respect to the cloud-based approaches ($\approx$ 25%-30%).

Advanced driving: platooning: A key ITS application that will benefit from the presence of MEC is platooning [130]. The latter is a paradigm that allows a group of vehicles to drive together, in line, decreasing the distance between vehicles. This allows to increase the number of vehicles on the roads without incurring in traffic jams, to augment safety and to save money on fuel, thanks to the drag effect, thus limiting the overall emissions. Platooning requires to have very low latency because vehicles can travel several meters in a fraction of second and a fast access to computing capabilities. Figure 4 shows an example of platooning leveraging on a possible MEC deployment whose applications to platooning has been recently proposed in a few works. As an example of system design, Montanaro *et al.* [131] present a 3-tier architecture for controlling and managing platoons of vehicles using cloud and edge computing capabilities. Furthermore, the authors of [132] propose a MEC architecture with the goal of avoiding shock waves, for instance due to asynchronous brakes, during platoon driving. As an example of computing opportunities offered by platooning, the authors of [133] study the offloading decision of collaborative task execution between a platoon and a MEC server, with the goal of minimizing task offloading decisions. In [134], the authors provide a framework where the MEC performs a platoon formation and coordination algorithm, receiving periodically updates from vehicles on speed and position. They show that their algorithm achieves low computations and delays in a realistic LTE-Advanced simulation scenario. Finally, Quadri *et al.* [135] state that a MEC centralized control of speed and acceleration of platoon vehicles is a viable alternative to common distributed approaches such as V2V communications. Through a detailed Python simulator they show how, notwithstanding the impact of delay and packet loss probability caused by new UL/DL communications towards the RAN, a MEC centralized control of platoons in 5G networks will help in reducing fuels costs (i.e., allowing smaller inter-vehicle distances), while at the same time supporting a large density of vehicles without incurring in congestions.

Vehicular clouds: Several works propose to move the computing resources within the ITS users. Zhang *et al.* [136] propose a hierarchical cloud-based vehicular edge computing (VEC) offloading architecture, with the goal of reaching the optimal computation offloading, considering both the minimization of task delays and the maximization of network provider's revenue. In [137], the authors propose a collaborative MEC scenario depending on the so-called heat zones, where different degree of heats stands for vehicles density inside a certain area, for vehicular task offloading. In order to achieve the MEC cooperation they formulate it as a utility maximization problem by designing a non-cooperative game-theoretic strategy. Through simulations, they show the feasibility of their solution comparing it with several policies.

In [75], the authors propose to use FMeC for handling computing problems with the computing power of vehicular clouds (see Section III for further explanations). Other works focus on the possibility to use the vehicles themselves to create a (micro) cloud. In particular, this branch of research is often referred to as *vehicular cloud computing*. The definition was first proposed by Gerla [138] and it has been further developed in [139]. With the vehicular cloud computing paradigm, a user sends a request to a car using V2V or V2I communications and then the request is forwarded to discover a communication path to a vehicle offering the desired computing service. Afterwards, data exchange and computation happen. For instance, Copeland *et al.* [140] describe the AVEC paradigm (automotive virtual edge communicator), which leverages computing resources and advanced technologies that could be present inside vehicles and that could be exploited during emergencies. In [141], Dressler *et al.* leverage parked cars as edge network and storage infrastructure, forming therefore a *vehicular cloud*, with the goal of boosting the performance and scalability of vehicular networks. In this scenario, they propose a protocol called *virtual cord protocol*, in order to sustain the dynamic of this scenario (with cars that can come or leave) and show that their protocol is able to sustain this scenario. Hagenauer *et al.* [142] introduce the concept of vehicular micro clouds (a cluster of cars acting as virtual edge servers). These clusters aggregate all the data which are then transferred to the data center in the cloud. In their paper, they propose a map-based clustering, which is then evaluated against different aggregation rates and backhaul technologies. In another paper [143], the same authors deal with two major problems given by this infrastructure: the selection of the gateway nodes and consequentially, the handover procedures. In [144], Dressler *et al.* propose a novel approach called *macro-micro-cloud*, with the goal of reducing the communication complexity and to improve the QoS, exploiting an additional layer, called *virtual edge computing layer*, between the data center placed at the core of the network and the users which should use the MEC features. This part is called macro cloud, while the micro clouds are clusters of cars.

Infotainment: Finally, some examples on vehicular infotainment are provided. Ndikumana and Hong [145] propose to serve self-driving cars by deploying MEC resources at macro base stations, Wi-Fi access points, and roadside units for caching infotainment contents in close proximity to the customers. The same authors, in [146], propose infotainment caching in self-driving cars, where caching decisions are based on passengers' features obtained using deep learning, showing that their approach reduces the backhaul traffic by 61%.

B. IoT and Smart City

IoT: IoT takes under its umbrella all the devices that can connect towards the Internet. Some examples are UAV, devices for home automation such as lighting, fridges but also Alexa, Google Home, medical devices and manufacturing devices. They all need different requirements in latency, storage, bandwidth and security [147], and to support the new possibilities given by 5G such as network slicing [148]. Therefore, the advent of the MEC paradigm seems perfect to help IoT meet all its requirements, as discussed in [149]. However, our survey does not focus on the MEC support for general IoT, since in literature several surveys already cover this topic (see [10], [13] and [18]). We rather focus on new MEC-enabled verticals, one of which is smart city.

Smart City: Indeed, thanks to the increasing importance of the IoT paradigm, also cities are now evolving, installing sensors and IoT devices and therefore becoming *smart* [150]. The collection of data from users, IoT devices, sensors or from more generic devices will allow to understand deeper which are the critical points of a city management and therefore to help developing new strategies, with the goal of reducing costs, improve safety and resource consumption. Furthermore, projects such as SmartSantander [106], 5Gcity [151] or SynchroniCity [150] are giving a glimpse of what the city of the future will look like. According to [150], five macro-themes are currently evaluated in most of the smart cities:

- *Mobility:* this includes smart and secure car/bike parking, electric bike usage monitoring, public transportation usage, traffic optimization, adaptive lighting.
- *Sustainability:* some examples are noise pollution planning, air quality evaluation, urban waste management, water management (also called Smart Water).
- *Governance:* for instance, agile governance, environment monitoring, open data accessibility, citizens engagements on urbanization.
- *Data Mining:* data lake value extraction.
- *Security:* citizens awareness of IoT.

MEC implementation: MEC seems the most promising technology to sustain the smart city paradigm. Indeed, thanks to its *multi-access* paradigm, it will support the connectivity of a variety of devices (GPRS/UMTS/LTE, Wi-Fi, or wired interfaces) all together. Moreover, it can collect and real-time process, for instance, large amounts of data, and store local information (for security purposes), thanks to its deployed physical edge capabilities. Thanks to the low latency achieved by the MEC presence, a driver could then be informed in a very short time if an accident happened somewhere in the city and which alternatives, he/she could take. Similarly, cameras can perform a first processing of the recorded images at the edge, sending the frames to a central cloud only for special purposes.

Even though the literature on MEC in smart cities is still scarce (most of the papers are magazines), it is possible to draw some directions on the ongoing research efforts. Several papers tackle the issues of MEC implementation in smart cities or even in smart homes, the latter leveraging on D2D communications [152]. For instance, in [106], the authors propose a MEC architecture for large scale IoT deployments (as Smart Cities) supporting existing and future IoT platforms and compliant to the ETSI MEC standard. The authors of [153] propose a smart city scenario in which real time and time-sensitive applications offload their tasks to MEC servers deployed in cars. They propose an optimization problem to minimize the completion time with a given cost of task scheduling, developing four evolving task scheduling algorithms. Through simulations, they compare them against each other, highlighting that only one (the distributed and improved Jacobi ADMM algorithm) reaches performance close to the optimal.

Machine learning: Furthermore, in smart cities it is important that decisions at MEC level are fast and mostly correct. Machine learning, especially in the form of deep reinforcement learning (DRL), seems a promising solution to achieve these goals. In [154] the authors propose a framework which leverages SDN, ICN and MEC computing capabilities to provide caching and dynamic orchestration of computing resources at the edge. Their goal is to improve the performance of applications in Smart Cities. They developed a big data DRL algorithm and through simulations they show the higher performance of their solution in terms of total utility (up to 60%) compared against several schemes (e.g., same scheme but without edge caching or virtualization etc.). Liu *et al.* [155] state that green energy management systems are becoming more and more important due to the development of smart cities. Hence, they develop a model for IoT-based energy management system, leveraging on DRL, on top of a edge computing infrastructure. They compare their solution in terms on delay and energy cost against baseline energy scheduling methods (e.g., only-cloud methods), showing that their DRL-bases method achieves less energy cost (up to 60%) and a smaller overall delay (25%).

Zhao *et al.* [156] study the always changing service demand due to crowds in a Smart City. To balance the network load and avoid network congestion and annoying delays, they develop a smart algorithm based on DRL, showing that it achieves better performance than algorithms such as OSPF and EOSPF (from 10% and up until 50%, on average).

Video streaming: smart cities themselves will also be a container where other verticals (e.g., automotive, media, manufacturing among the others) will be merged and further studied. However, in this context, only media has been evaluated in smart cities with MEC so far (especially for video streaming). In [151], the authors explain the 5Gcity project, which has the goal to create a MEC neutral host platform for smart cities, focused especially on ultra-high definition video streaming, live streaming and AR/VR use cases. To show the feasibility of their architecture, they evaluate three different use cases by deploying testbeds in three European cities (Bristol, U.K.; Lucca, Italy; and Barcelona, Spain), In another paper that tackles video streaming in smart cities, Taleb *et al.* [157] propose the merge of FMeC concepts (evaluated in Section III) with MEC capabilities in order to maintain constant the QoE of video streaming while users move. Specifically, they enable MEC service migration to follow users.

Security: Both security & privacy are topics of uttermost importance for smart cities. Indeed, the collecting, management and processing of sensible data at the edge could lead to attacks from malicious user or to data breach, with catastrophic scenarios. The next discussed group of papers focus on several aspects of security with MEC in smart cities.

In [158], the authors propose a selective recommendation mechanism based on compiling dynamic black- and white-lists, so as to identify trustworthy participants that can access smart city devices. With data-driven experiments, based on both personal health and air quality monitoring, they show the effectiveness of their solution in avoiding malicious attacks in various scenarios, comparing it also against other similar algorithms proposed in [159] and [160]. Wang *et al.* [161] focus on the security threat given by low cost IoT devices and the MEC deployment near the RAN. They state that upper layer cryptography is not feasible for resource-limited scenarios and propose a comparison between information security mechanisms implemented via physical layer approaches. Rahman *et al.* [162] propose a framework that leverages blockchain, AI and edge nodes in order to offer secure smart city services (sharing economy, smart contracts, and cyber-physical interaction. Finally, Gheisari *et al.* [163] propose a privacy-preserving architecture, leveraging on ontology at the edge network, for IoT devices in a Smart City scenario. Through simulations, they show that the ontology would allow for preserving privacy in a heterogeneous IoT scenario.

C. Media

As of today, 70% of the overall data traffic owns to video applications, e.g., it comes from platforms like *Netflix*, *YouTube* and *Twitch*. In the next years, this share is expected to grow due to the advent of virtual and augmented reality applications. These applications impose tighter constraints than other video applications, especially in terms of delays, bandwidth and computation [164]. Therefore, both for canonical video streaming and AR/VR, it is of vital importance to move resources at the edge of the network, leveraging on the new MEC paradigm.

Video streaming: In this context, MEC will be useful for increasing the overall Quality of Experience (QoE), exploiting several approaches such as caching, cooperation between MEC nodes and offloading of heavy computational tasks (e.g., transcoding), even merging these concepts together. For instance, Tran and Pompili [165] propose to leverage collaborative MEC servers to enhance video caching and processing support for adaptive bit rate (ABR) video streaming. This collaborative joint caching and processing problem is formulated through an integer linear problem, with the goal of minimizing the average access delay to video users. To address this problem, they formulate a low complexity online request. They use simulations to show that their approach outperforms by $\approx$ 20% caching techniques such as Most Popular Caching and other schemes [166]. The authors of [167] study the caching at the edge for improving the QoE of live video streaming. They propose two auction frameworks for the caching space allocation at backhaul (Edge Combinatorial Clock Auction and Combinatorial Clock Auction in Stream), showing via simulations that they achieve higher performance, about 10% better if compared to baselines.

In [168], the authors design a scenario for video streaming with MEC resources, studying how fairness (of edge computation capabilities) and QoE can be improved with MEC against baseline client-based DASH heuristics. Using a network simulator (SimuLTE) they show the superiority of their scenario in terms of bitrate per client (20% higher on average), initial buffer delay ($\approx$ 15%-20% smaller) and Jain's fairness index [169]. The goal of Long *et al.* [170] is to improve detection accuracy of human presence using cameras. They leverage cooperative MEC nodes for pre-processing tasks. Their focus is especially on how to partition video tasks and how to match tasks to edge nodes. The MEC, thanks to the its edge computing resources, can exploit tools such as machine learning and blockchain to support QoE improvements. The authors of [171] propose a proof of concept based on LTE for MEC support to mobile video streaming. The MEC server caches popular videos and, based on the radio condition, chooses the most suitable video quality. They further propose two machine learning algorithms for popular video prediction and forecast of channel quality. Through numerical simulations, they show, for instance, that the prediction model for radio channel quality reaches over 80% of prediction accuracy. Instead, Liu *et al.* [172] propose a blockchain video streaming framework assisted by MEC, where heavy computational tasks such as video transcoding can be offloaded to MEC nodes. They compare their solution against the same one without the blockchain component, showing that the latter perform worse, up to 35%, in terms of average delay.

Finally, several papers tackle MEC implementation with real LTE testbeds, to support video streaming. Martin *et al.* [173] design a new MEC component for video streaming called MEC4FRE. This application retrieves data analytics from layers 2 (RAN awareness), 3 (media delivery metrics) and 7 (MPEG-DASH manifest for local caching) in order to dynamically prevent QoE degradation and keep radio efficiency high. The authors compare their solution against a best-effort delivery strategy in a real LTE infrastructure, where they proved that their solution achieves better performance.

Ge and Wang [174] present a novel MEC real-time QoE estimation VNF, which has been implemented and deployed in a real LTE-A network edge. They show that their VNF is able to correctly estimate QoE in real time and its CPU and RAM usage are both very low.

AR/VR: Thanks to the recent technological hardware advancement, more and more realistic Virtual Reality (VR) and Augmented Reality (AR) applications are present, notwithstanding the demanding bandwidth and delay requirements. According to Huawei Technologies and the China Academy of Information and Communications Technology (CAICT) [175], in order to achieve the entry level immersion experience in VR, with a 4K 2D video, the bandwidth provided to the service should range between 20-50 Mbps with a round trip time (RTT) latency of maximum 40 ms. Instead, for a fully immersion experience (with a 24K 3D screen), the bandwidth should range from 2 to 5 Gbps and RTT below 10 ms.

Indeed, according to [176], MEC features such high proximity computing, proactive caching and support to mmWave are needed for AR/VR successful delivery, taking into consideration also that computing and communications delays are the two most relevant bottleneck in AR/VR cases. Hence, a MEC deployment becomes of primary importance. The authors of [177] propose an integrated heterogeneous networking scheme, taking into consideration the fiber-wireless access networks, using a virtualization techniques to achieve the demands of the applications. They evaluate their solution with a testbed, showing that this infrastructure supports the AR/VR requirements and outperforms other paradigms such as Mobile Cloud Computing in terms of RTT latency (with differences up to 50%). In [178], the authors define the main challenges for a full wireless interconnected VR (Quality-rate-latency tradeoff, Localization and tracking accuracy, green VR among the others). Further they focus on three possible interconnected VR study cases: the first is about leveraging on the joint resource allocation and computing, the second one shows the benefits stemming from exploiting proactive computing against reactive computing, while the last one studies an AR enabled case with self-driving vehicles. With simulations, they show that with nowadays technologies it is still impossible to reach a fully interconnected VR scenario.

Similarly, the authors in [179] argue that most of the works in this area consider only computation-constrained MEC scenarios, neglecting the communication prospective. Therefore, they propose a MEC framework with the goal of reducing the communication-resource consumption leveraging caching and computation resources of VR devices. They formulate an optimal task scheduling policy to minimize the average transmission data per task. Through numerical simulations, they show that it achieves higher performance in terms of average communication costs ($\approx$ 45%) compared to baselines.

Immersive videos for VR, also known as **360 degree** videos, provide an interesting VR feature, thanks to their omnidirectional view they offer. Several papers tackle the use of MEC for immersive videos. Liu *et al.* [180] develop a multi connectivity scenario for 360 degree videos (MEC's computing resources for active transcoding and caching + mmWave/sub 6 GHz for supporting high bandwidth VR). Furthermore, within their scenario they formulate a novel communication and computation resource allocation problem. Through simulations, they compare their solution against cases in which some technologies were not present, showing that it achieves better performance in terms of latency and energy efficiency (from 15% up to 25% on average). In their paper, Sun *et al.* [181] model several trade-offs between communications, caching and computing with MEC in a mobile 360 degree VR scenario. They first propose a novel MEC framework for this scenario and then formulate an optimal joint caching and computing policy with the goal of minimizing the average transmission rate, under several constraints (latency, cache size and average power consumption constraints). They obtain a closed-form expression and evaluate it against several greedy algorithms, showing that it achieves higher performance (depending on the scenario, from 30% to 50%). Mangiante *et al.* [182] propose a rendering solution for 360 degree videos leveraging on MEC, with the goal of optimizing the latency and bandwidth resources. Through preliminary tests, they show the benefits of having an edge network infrastructure in terms of reducing by up to 80.5% data traffic delivered towards a centralized cloud and radio access.

D. Manufacturing

In 2019, the 5G Alliance for Connected Industries and Automation (5G ACIA)[23] was created. Its goal is to apply 3GPP 5G specifications for Smart Factories [183] to the operation of manufacturing and processing industries. 5G ACIA has six working groups, covering aspects like architecture and technology for industries, use cases and requirements and spectrum and operating models among the others.

Smart factories are context-aware systems that "assist people and machines in execution of their tasks" [184]. The context includes the status and position of an object both based on virtual and physical information available, enabled by both machine-type communications and IoT devices.

The MEC is an important means towards implementing some of the key design principles introduced with the Industry 4.0 paradigm. In particular, it paves the way towards interoperability of machines, virtualization of physical resources, decentralization and real time capabilities in the analysis of data (thanks to the support of VNF, 3rd party and industrial applications). The use of MEC also helps in terms of achieving low delays, which is vital for some IIoTs applications that tolerate no more than 250 μs delay [185] (such as robot motion control and packaging machines).

In real case examples, the MEC might have access to all the processes in a Smart Factory, from logistics to supply and inventory management. The MEC might therefore be able to retrieve data from all the sensors of IIoT devices, and automatically and dynamically make decisions according to a predetermined goal.

MEC infrastructure: Due to the diversity and complexity of factories in terms of production, machinery, spaces and specialized workforce, the MEC infrastructure needs to be carefully designed in order to allow the proper level of flexibility for smart manufacturing plants. A first attempt to provide a specific MEC infrastructure for smart factories is in [186]. The authors propose a 3-level hierarchical smart factory architecture, in which they highlight a physical resources layer, a network layer and a data application layer. The first layer contains all the manufacturing resources that, through sensors and RFID (among the others), can interact with the

[23]https://www.5g-acia.org/

second level. The latter includes networking technologies such as access points access and switches (deployed according to new paradigms like MEC and SDN). Finally, the third layer allows for the analysis of the retrieved data in order to gather useful information about the status of the smart factory, to be sent to end users (workers or engineers). Similarly, Dao *et al.* [187] propose an mMEC, i.e., a multi-tier MEC architecture keeping in mind several IIoT challenges such as the processing of big IIoT Data with ultra-low latency and reliable response, and context awareness. Finally, the authors of [188] propose a hybrid computing solution framework, with the goal of proposing a resource scheduling strategy for real time smart manufacturing applications in a edge computing scenario. Through a prototype implementation, they show that their strategy outperforms other approaches, e.g., centralized cloud, in terms of computing latency ($\approx$ 15%-33% on average).

Reliability: This is a topic of uttermost relevance in smart factories. IIoT devices need $>$ 99,999% of successfully transmitted packets, in order to avoid malfunctioning in the production lines and accidents that could harm workers. The following papers provide the most relevant examples of issues that impact the overall reliability of a MEC system in smart factories. The authors in [189] study a resource request banker's algorithm in order to avoid deadlocks that could occur in the presence of several IIoT devices accessing the MEC resources (a behaviour that they confirmed through simulations). With their algorithm, they prove that the probability of a deadlock in a MEC scenario will be reduced up to 12% compared to a scenario without any deadlock avoidance algorithm. Luo *et al.* [190] propose an adaptive task offloading auction mechanism that allow Industrial Cyber Physical Systems (ICPS) to offload their tasks to several MEC servers chosen based on task deadlines and the required security levels. By means of simulations, they show the superiority of their approach compared to baseline schemes using randomized and FIFO scheduling. Finally, the authors in [191] propose a 2-tier partial offload MEC-cloud framework in a heterogeneous energy constrained IIoT scenario with the goal of optimizing the transmission reliability and IIoT energy consumption. They formulate a low complexity solution and evaluating it through simulations. They compare their algorithm against two baseline solutions, showing that it achieves higher performance in terms of energy consumption and blocked devices (from 10% to 20%).

E. eHealthcare

Another important vertical which is gaining attention is eHealthcare. Medical tools are becoming more and more sophisticated, with multiple sensors and data (ranging from video, signals and personal) that has to be processed. Moreover, consumers are paying progressively more attention to well-being, with an increasing demand of quality devices, safety and data storage. Therefore, these requirements bring the necessity to move computational resources closer to devices, in order to perform faster, efficient [192] and accurate decisions.

Edge nodes can also be leveraged for performing data preprocessing, in order to send only selected data towards a centralized cloud, helping in both reducing bandwidth utilization and improving privacy. On this line, the authors of [193] study an abnormal pattern detection mechanism of a patient's state at the edge of the network, where edge nodes send only the most important features in a centralized cloud. Further, in case of detected anomalies on the patient's state pattern, it pings the nearest healthcare provider. In another paper, the same authors enhanced the framework proposed earlier with the MEC architecture [194], highlighting the benefits that MEC will bring in several smart health applications (for instance low latency for real-time epileptic seizure detection or prediction of bradycardia in preterm infants or reducing bandwidth allocation for continuous services such as remote cardiac monitoring or Parkinson's disease detection). Similarly, in [195], the authors leverage the MEC for a preliminary data processing of electroencephalogram signals (for smart pathology detection) before sending the data to a centralized cloud. Pace *et al.* [196] propose to create an edge layer between cloud and IoT devices belonging to end users, with the goal of reducing communications delay and increase privacy level. They evaluate their framework with a real test bed in two different scenarios (workers in a factory and athletes in a fitness center), showing that their framework would reduce the communications delay and the overall data transmitted to the centralized cloud by 20%–50%. In [197], the authors propose to collect health information to monitor patient's health via UAVs and then processing the data in MEC servers (possibly in the nearest one) leveraging blockchain for increase data security. Through simulations, they show the effectiveness of their scenario. Chen *et al.* [198] describe a cognitive edge computing smart-healthcare system, with the double goal of evaluating the patient's health using an edge cognitive computing paradigm and, depending on health-risk grade of each patient, allocate edge communications resources to better assist them in emergency situations. Furthermore, in [199], Muhammed *et al.* propose a framework called UbeHealth, which leverages edge computing, deep learning, big data and high-performance computing to support healthcare systems in smart cities. They developed a proof of concept and performed an evaluation based on a nationwide networked healthcare system with three different data sets. They show that, with their proof of concept, latency is reduced by 50% compared to cloud-based healthcare solutions. Finally, Li *et al.* [200] present Edgecare, a secure and efficient data management system, with the goal of improving the management of decentralized healthcare data, leveraging edge computing paradigms such as MEC. They propose an optimization problem and, through numerical simulations with security analysis, showed the effectiveness of their framework.

F. Summary, Lessons Learned and Open Challenges

In general, so far MEC has not been fully evaluated for vertical industries. Most of the papers reviewed in this survey are architectural, with few of them that analyze real datasets or evaluate performance figures of real devices. However, it

TABLE IV
LIST OF PAPERS SURVEYED IN SECTION IV

Industrial verticals	References	Use Case	Analytical tools	Evaluation	Most relevant lessons learned
Automotive	[127], [130], [126] [128], [129], [138] [137], [124], [76] [141], [142], [143] [144], [145], [132] [133], [135], [146] [147]	- Safety - Avoid collisions between vehicles and vehicles/ Vulnerable Road User - Advanced driving: platooning - Computation offloading - Vehicular clouds - Infotainment	- Models - Fuzzy logic algorithm - Collision avoidance algorithm - Network model - Graph theory - Stackelberg game - Optimization	- Numerical simulations - Network simulations	- MEC hosts reduce latency up to 80% compared to common network architecture - Integrating several access technologies together (sub-6 GHz band, mmWave and IEEE 802.11p) improve performance in highly dense scenarios with low bandwidth - Autonomous cars detect 100% of collisions. With human drivers, the number slightly decrease by 14% - Using Deep Learning for infotainment caching reduces backhaul traffic by 61%
Smart City	[107], [155], [152] [156], [157], [154] [163], [159], [162] [164]	- Augment ETSI MEC standard to support Smart City - Smart Home - IoT-Based energy management in smart cities - Access service rate for big crowds - Task scheduling for smart city applications - Blockchain for sharing economy services - Security threats on physical layer - privacy preserving	- Big Data - Deep Reinforcement Learning (DRL) - Optimization - Lagrangian function - Artificial Intelligence - Lyapunov theory - Data model (Ontology)	- Numerical simulations - Testbed simulations - Trace-driven simulations	- Cooperative DRL (leveraging both cloud-edge resources) reduces delay up 25% and energy cost up to 60%, compared to an only-cloud based solution - DRL can also achieve higher service access rate for big crowds compared to baseline solutions (OSPF and EOSPF) - Physical layer security can be added in a heterogenous IoT scenario thanks to it low complexity and resource allocation
Media	[166], [169], [173] [168], [174], [172] [175], [178], [183] [179], [180], [182]	- ABR video streaming with MEC - Cache placement - Block chain video streaming assisted by MEC - QoE enchanments - Cooperative video processing - AR/VR support	- Integer linear programming - Optimization - Auction frameworks - Dynamic programming - Optimal matching theorem - Multipath routing algorithm - Lyapunov theory	- Numerical simulations - Network simulations - Testbed performance evaluation	- Caching with MEC will improve backhaul traffic load and average access delay with respect to established approaches - MEC, together with fiber-wireless access networks, will outperform the Mobile Cloud Computing paradigm in terms of RTT latency (up to 50% of difference) - MEC could support VR in in terms of latency reduction (compared in scenarios w/o MEC) and energy efficiency - MEC processing of VR tasks will decrease the traffic in core and radio access up to 80.5%)
Manufacturing	[187], [189], [188] [192], [190], [191]	- Resource scheduling for low latency services - multi-tier MEC architecture for satisfying several IIoT requirements - task offloading in smart factory - Avoiding deadlock in resource provisioning	- AI - Two-step algorithm - Deadlock avoidance algorithm	- Prototype evaluation - Numerical simulations	- MEC's proximity will decrease computing delays (up to 40%) and energy consumption
eHealthcare	[194], [196], [199] [198], [195], [197] [200]	- Abnormal patter detection in patient's state - Comprehensive MEC architectures for smart health - EEG- based pathology detection system - Blockchain based health monitoring	- Feature extraction - Signal processing - Tree-based deep model - Bloom filter - Deep Learning - Stackelberg game optimization algorithm	- Numerical simulations - Testbed experiments	- MEC will help health applications in a wide range of fields, from Data reduction, bandwidth and energy saving and low latency - MEC will reduce the latency up to 50% compared to cloud-based networked healthcare systems

seems reasonable to predict that this gap will be closed soon. This is because of the newly deployed testbeds[24,25] and on the roll-out of 5G systems, which has been already started. Looking at a more general prospective, Table IV shows that the most studied verticals are automotive, smart city and media. With the help of the table, we next comment we comment on

[24]http://5g-transformer.eu/
[25]https://www.openedgecomputing.org/

lessons learned and open research challenges for each of the verticals.

Automotive: Focusing firstly on the automotive domain, we see that MEC is considered a fundamental building block for achieving efficient C-V2X communications, thanks to its possibility to achieve low latency [120], [121]. The main takeouts can be summarizes as follows:

- 5GAA has identified four possible use case groups: safety, convenience, advanced driving assistance and VRU. Important research efforts have been devoted to safety and VRU ([125], [126], [127] [128], [129]), showing that the MEC presence, thanks to its proximity to the end users and high computation power, will be of great help to improve both vehicle and pedestrian safety (for instance by offloading the computation of collision detection algorithms to close MEC servers). Indeed, according to [127] 100% of collisions with autonomous cars could be detected on time, while with human drivers the number slightly decrease by 14%.
- However, offloading decisions are not trivial to make, since they should also consider the presence of a possibly high density of vehicles [137], and revenues generated by different vehicles [136].
- MEC will also help to provide infotainment to drivers and passengers ([145], [146]), especially leveraging caching together with deep learning, which allows reduce backhaul traffic by 61%.
- Most of the available papers have identified a number of technical challenges, such as enabling edge communications ([123] proposes to use three different access technologies), or ad hoc computing resources such as vehicular clouds [140], [141], [142], [144]. The vehicular cloud paradigm allows to have computing resources even within vehicles, pushing the MEC paradigm at the very edge. This scenario however imposes tough challenges due to its volatility (for instance, a car might join the cloud any time). Specifically, data management and communications between clouds and backhaul (both in uplink/downlink) becomes cumbersome, needing therefore more in-depth research effort.
- Furthermore, it emerges that the MEC also supports pioneering assisted-driving applications, such as platooning. Several papers addressed this topic, showing that MEC offers a possible solution to sustain this paradigm. For instance, [131] and [132] propose an architecture for managing platoons and avoiding shockwaves, [133] focused on offloading decisions, [134] proposed a MEC that can form and coordinate platoons. Finally, [135] showed that a MEC centralized control of speed and acceleration of platoon vehicles is a viable alternative to V2V communications.

Notwithstanding the large amount of published papers, still many open research challenges remain, e.g.,:

- **Security** is the uttermost theme to be developed in vehicular networks assisted by MEC. Indeed, with the growing possibility of having more connected cars and edge resources on the road, there are also more possibilities for malicious attacks. These scenarios must be avoided and therefore research should focus more on the security aspects of these new paradigm that embraces connectivity and computation. Some examples of risky procedures in the MEC environment are migration of resources (VMs and Containers), MEC deployment billing, and what refers to the coordination of multiple new nodes introduced with the MEC architecture [111].
- More work should also be oriented to **VRU** and general **safety**, with the development of new collision detection or avoidance algorithms, leveraging also on prediction techniques given by machine learning which take into account both physical resources and wireless channels [201].
- According to Intel,[26] a single autonomous car could generate up to four terabytes of data each day. Hence, the MEC should be able to handle and process that amount of data. Which is more, the MEC should support multiple autonomous cars at the same time. Therefore, **big data** processing and analytics is of fundamental importance for both connected cars and MEC paradigms. However, no work has so far addressed jointly these issues in a vehicular scenario.
- With the possibility to deploy computing resources at the edge, new **business** opportunities arise, together with the possibility to increase **revenues**, in multi-operator scenarios [124]. While [136] provided a first example of a possible MEC-based revenue generating system, more research is needed to fully cover the complexity of this scenario.
- Many examples provided by 5GAA [124] and [202] have not been studied and evaluated yet: some examples are real time situational awareness and handling high definition maps. Moreover, future work should also consider 5G features such as network slicing, or the support to new Internet architectures, such as ICN [203].
- Finally, the car manufacturing world is slowly shifting from traditional oil-based vehicles to **electrical/hydrogen** vehicles. It would be interesting to study how and if this shift would also affect the MEC support for vehicular networks, and if the MEC could play a role in making cars *greener* and more efficient, and smarter in general.

Smar city: While IoT as a macro concept has been widely studied, what it has not been fully explored yet is the MEC implementation in smart cities, where the MEC can play a fundamental role for the communication part. Indeed, while the smart city paradigm embeds different verticals (e.g., automotive, Media, eHealthcare) all together, it poses new challenges and constraints due to its enhanced IoT deployment nature. In the SmartSantander case, more than 20000 sensors (between fixed, mobile and smartphones ones) and 2500 RFID tags [106] have been deployed, posing therefore scalability and QoS challenges (for instance, how to avoid that collisions between packets coming from hundreds or thousands of devices would degrade the throughput significantly). Below are listed some lessons learned and open research challenges:

[26]https://newsroom.intel.com/editorials/krzanich-the-future-of-automated-driving/

- First of all, most of the papers surveyed are magazines. While they give a great overview on most of the possible technical scenarios for smart cities, they lack on in-depth technical view, which instead is needed to better study this vertical.
- Several papers identify the need to define new framework architectures to support this vertical. Reference [106] proposed an architecture compliant to the ETSI MEC architecture, to support enchanted IoT deployments, [154] merged MEC, SDN and ICN for caching at the edge while [151] showed the 5Gcity project, aiming to develop testbeds for UHD video streaming in smart cities.
- Many papers claim that optimization techniques and machine learning are keys to finally deploy smart cities ([154], [153], [155], [156], [162]). Both [155] and [156] show that using deep reinforcement learning for energy management systems at the edge and network load balancing in the presence of moving crowds can outperform traditional approaches (e.g., only centralized cloud methods) and algorithms (such as OSPF and EOSPF) from 10% up to 60%. [153] shows how optimization problem for offloading tasks are crucial for real time-sensitive applications and [162] unveiled the advantages of blending blockchain, AI, and edge computing to support sharing economy services.
- Finally, several papers point out that severe security issues are unresolved. Reference [158] shows that selecting trustworthy participants for accessing smart city services is desirable, while [161] points out that the main information security challenges are in the physical layer. Reference [163] warns against the lack of suitable privacy preserving mechanisms.

Indeed, while the concept of smart city has been theorized many years ago, more technical work is still needed to make it real:

- As for the other verticals, increasing connections between users and things in a smart city context gives hackers the possibility to obtain important personal data, both directly (social security numbers, bank accounts etc.) or indirectly (by inferring political or religious preferences, etc.). Attackers could leverage the weaknesses of the network infrastructure. Hence, more comprehensive work on **security** and **privacy** issues should be performed, maintaining both a full stack overview and aiming at lightweight solutions, which could be deployed on simple objects with the help of the MEC. Furthermore, we believe that research should specifically consider security physical attacks such as power cutout, fire and link break [111] (due to the presence of a high population density scenario) creating a more resilient distributed system.
- **Machine learning** would be a useful tool to predict crowd/vehicles movements or network traffic, e.g., to avoid congestion. Among all the techniques, federated learning seems the most promising one to preserve users privacy, since it allows the decentralization of data by only exchanging encrypted machine learning parameters between edge nodes and a centralized server. Similarly, as already discussed in Section III-D, **Green MEC** systems should be considered in order to reduce costs for operators or even for public administration entities.
- While the papers surveyed cover a quite wide spectrum, however many real use cases are still unexplored [150]. Some interesting examples are how to deal with waste management in real time through smart grids, the use of smart management, or also the synchronization of traffic lights given the presence of crowds and vehicles, leveraging also on ML techniques. MEC, thanks to its computing capabilities and user proximity, will be a possible enabler for these use cases. Researchers should also consider merging several verticals, like media or automotive, together with smart city in order to provide a more realistic scenario.
- Most of the works are theoretical: it would be interesting to leverage real testbeds such as AWS Green Grass or Azure IoT Edge in order to compare the performance in real case scenarios.
- Researchers should also address these fundamental problems: how to provide scalability with a high number of IoT devices (the SmartSantader project deployed more than 20000 sensors), interoperability between several propriety interfaces (for instance how to allow communications and cooperation between AWS Green Grass, Azure IoT Edge and the ETSI MEC framework), study how to develop new business models (since this scenario gives new revenue opportunities to operators), or how to support network slicing for multiple tasks or verticals on a shared MEC server/infrastructure or support smart caching at the edge.
- As a possible smart city sub-case, MEC together with smart homes has not almost been evaluated yet. Nowadays, our homes are welcoming more and more “smart" devices (e.g., TVs, home automation devices, vocal assistants such as Alexa and so on). MEC would help those appliances in several ways: from contents cached in close MEC servers in order to improve QoE, to IoT Data pre-processing at the edge (leading to less information sent over the Internet, with implications on users’ privacy), to a support of and integration in the local smart grid.
- Finally, an interesting paradigm in smart cities are UAV communications with MEC. Indeed, UAVs (or commonly known as drones) are becoming more and more powerful while at the same time their costs are decreasing. Nowadays, UAVs are exploited in many different fields ranging from weather monitoring, precision agriculture, to package delivery and traffic control [204]. Therefore they are also evolving the concept of smart city into a bigger smart “metropolitan” area (see Section V). MEC together with UAVs enhance computing offloading at the edge (with a UAV based MEC server that compute users tasks) or help UAVs themselves during heavier computing tasks (particularly helpful since in most of the cases UAVs batteries have a limited battery life) [18]. Finally, MEC can exploit the O-RAN architecture to better support UAV communications (e.g., to allow radio resource allocation

for UAV Applications or flight path based dynamic UAV resource allocation [41]). For interested readers, we mention more focused surveys related on MEC together with UAV communications ([13], [18], [204]).

Media: MEC will also help in the development of new reliable video streaming connections and in the improvement of AR/VR applications, which impose very tight requirements on bandwidth and latency.

The key lessons learned are:

- In order to improve the QoE of video streaming, a few approaches are beneficial: leveraging caching thanks to the new MEC computing capabilities ([165], [168], [167]), blockchain ([172]), cooperation between MEC nodes ([165], [170]), offloading of heavy computational tasks such as adaptively adjusting bitrate or transcoding ([165], [172], [173], [170]) and machine learning techniques to forecast the channel quality [171]. The MEC will help in improving performance from 20% up to 35%.
- Many works provide also insights on performance within real LTE infrastructures ([173], [171], [174]), showing that the MEC presence, even just in LTE architectures, will be beneficial in terms of QoE estimation to prevent degradation, mainly thanks to its computing capabilities at the edge.
- On the AR/VR side, papers point out the need for a novel architecture ([176], [177], [180]) able to manage resources in order to tradeoff performance, communications or computing capabilities, taking into consideration the highly demanding AR/VR requirements, against the however limited MEC resources [178], [179], [181], [182].

There are several open research challenges:

- Regarding video streaming, only few works addressed the **live** case, which imposes tighter requirements than classic video streaming. Live streaming websites such as *Twitch* and live video conferencing are becoming more and more important for the everyday user, especially in alert circumstances like the one generated by the Covid-19 pandemic, hence it would be interesting to dig more on how to improve the overall QoE, leveraging the MEC concept.
- While some works propose to use ML to forecast channel quality [171], the possibility to deploy an intelligent MEC node between the end users and a remote cloud server has not been fully evaluated yet. ML can help for smart caching, forecasting the video streaming load according to traffic patterns and smart transcoding, among the others, and therefore it will be useful in resource constrained scenarios.
- Most of the available testbeds use LTE. While 5G has not been fully rolled out yet, still 5G coverage spots keep appearing in many cities around the world; thus, it would be interesting to see more work relying on a real 5G infrastructures.
- On the AR/VR side, many works focus on the highly stringent requirements and performance tradeoffs ([178], [181], among the others), questioning whether edge/MEC solutions would actually be a possible enabler. The answer is still unclear: while it undoubted that for a fully interconnected VR the road is still long, for baseline AR/VR, the MEC is however helpful for some task offloading, transcoding and caching functionalities. However, current MEC solutions are quite limiting, also due to the fact that MEC resources should be shared among different tenants, not necessarily belonging to the same vertical. It is also interesting to notice that while being close to the end user is an important enabler. In fact, for latency reasons, tasks processing delays caused by high AR/VR task demands might be still a relevant bottleneck for MEC and AR/VR applications. Therefore, tradeoffs between the edge computing infrastructure and VR devices should be further evaluated [176] (see Section V for further considerations).
- Another important new sector is **cloud gaming**. While existing solutions are somehow limited so far (*Google Stadia, Nvidia Geforce Now*) due the stringent requirements of gaming streaming (for instance, bandwidth requirements ranges from 10 Mbps for 1080p to a minimum of 35 Mbps for 4K)[27], 5G and MEC proximity deployments to end-users will surely help this paradigm to grow in terms of introducing newly available bands and offering smaller latency. This would open new possibilities to researches (and to markets). One research challenge consists in enabling cloud gaming applications to leverage several access technologies at the same time to increase the overall QoS and QoE.

Manufacturing: Another vertical which would benefit from the MEC presence is the Manufacture. Indeed, IIoT devices require low latency communications, high bandwidth and computing capabilities, reliability and security and at the moment only edge computing can satisfy all these requirements at the same time [205]. Reviewing the literature, the key lessons learned are:

- Most of the papers show the need to design a dedicated multi-level edge infrastructure for supporting smart factories, considering different constraints such as big data processing [186], [187], resource scheduling strategies [188] and reliability [191]. Compared to cloud solutions for manufacturing and smart factories in particular, a MEC infrastructure will decrease the computing latency and energy consumption up to 40%.
- Other works show the need to make the MEC reliable for IIoT, to prevent deadlocks [189], and highlight how offloading to MEC needs to be make based on manufacturing task deadlines [190].

Many challenges still remain open:

- The 5G ACIA has provided several useful insights for 5G deployment in smart factories [206], [207]. Focusing on the many MEC-related challenges in a smart factory, for instance, the MEC should be able to address a heterogeneous scenario consisting of several IIoT devices,

[27]https://www.forbes.com/sites/tiriasresearch/2020/02/04/nvidia-launches-affordable-geforce-now-cloud-gaming-service/#773ef8e1588b

each one with different demand and requirements. As an example, the MEC should support at the same time motion control devices (requiring a latency of < 1ms), mobile robots (latency of 10-100 ms) and traffic for human-machine interaction (for instance through VR devices). Hence, it would be of fundamental importance to study (*i*) how the MEC could provide and manage at the same time different QoS constraints and (*ii*) its resilience when dealing with variable data traffic (such as bursts). A possible solution could be offered by leveraging network slicing for differentiating several slices according to the QoS required. Finally, while some work focusing on the manufacture vertical has recently appeared (e.g., [208] and [209]) it remains interesting to evaluate how this vertical can benefit from the ETSI MEC standardization process.

- Also in this case, ML could be useful to solve some issues such as the ones related to the allocation of MEC resources. Furthermore, researchers could exploit the **Green MEC** paradigm in order to propose energy-efficient solutions for smart factories.
- Another very important aspect of MEC applications for manufacturing is security. The latter is fundamental for keeping IIoT data integrity. Otherwise, attackers might induce to machine failure or product quality issues. Data confidentiality is also key, because industry secrets must be protected. Security and **safety** in smart factories are very much tied, since security breaches might cause malfunctioning of production lines and of products, which could potentially harm workers as well as customers. A survey on the most common security attacks in NFV and 5G systems can be found in [111]. **Reliability** is a complementary aspect, since IIoT needs >99,999% of successfully transmitted packets. For instance, this can be achieved by deploying several MEC servers in order to create redundancy of resources (like in cloud datacenters) and/or a communications-wise management system, able to avoid extensive packets collisions. However, these solutions must also be **cost-efficient**.
- 5G ACIA also suggests exploring possibilities to converge together many communications technologies (D2D, Wi-Fi, antenna AP, sensors, RFID) in order to avoid wireless congested scenarios. As a possible solution, they propose to converge MEC and 5G with the Time-Sensitive Networking (TSN) framework, whose goal is to deliver deterministic services via IEEE 802 networks for wired industrial Ethernet solutions.
- Finally, next research steps should also consider the novel architectures proposed by 5G ACIA and evaluate their solutions with **real data-driven** traces, in order to study MEC in real case scenarios.

eHealthcare: The advent of IoT devices is changing also the healthcare system, which now is becoming *smarter*. While it is true that it somehow overlaps with IoT, eHealthcare systems present unique features that can be exploited to design an effective MEC system. Vice versa, the MEC can be exploited to deliver unprecedented life-saver technologies. For instance, every patient might have his/her personal data processed independently and securely, and health alarms might be triggered in a reliable way, avoiding privacy intrusions and false alarms, which means that the MEC should be thought as a secure and robust system. In turn, the presence of computing resources at the edge would help in the development of more sophisticated health machinery, which includes the support for remotely-driven surgery (e.g., tactile-Internet-based tele-surgery systems). The key lessons learned by overviewing state of the art works are:

- Most of the papers have identified MEC potentials for data pre-processing, to avoid sending too much sensible data over centralized clouds and to decrease sensible delays, up to 50%, compared to cloud-based networked healthcare systems [193], [194], [195], [196], [199].
- Reference [197] showed that blockchain can be also exploited to increase the security level with eHealth devices while, with the same goal, [200] proposes to manage healthcare data in a decentralized manner thanks to MEC nodes presence. Finally, [198] showed how edge resources could monitor a patient's health and be allocated in case of emergency situations.

Many open research challenges are still open:

- As for the previous vertical, also eHealthcare can leverage ML techniques. Indeed, due to the presence of edge computing resources, ML algorithms can be trained and applied to, e.g., quickly detect symptoms of diseases from images, therefore helping doctors in their diagnoses. In particular, federated learning seems a promising paradigm for privacy-related issues, since it allows to maintain the data locally in multiple decentralized edge devices. **Blockchain** could also be used to add protection to personal data from malicious attacks, and to make *auditable* the logs reporting the operation of the health staff, as well as the actions of patients. This might help to ensure that good practices are followed and would allow to identify conduct responsibilities in case of health issues.
- Moreover, it is interesting to notice that security and privacy are of fundamental importance in this vertical. However, adding advanced cryptography levels also increases computing overhead for resource-constrained edge nodes. Therefore, careful tradeoffs between security, computing and use of communications resources should be evaluated. For an overview of possible security threats, please refer to [111].
- Future works such also address users/patients mobility, with all the challenges it brings (see Section III-B for an in-depth analysis of the mobility challenges).
- While some works provided experiments in a real healthcare infrastructure [199], with the advent of even more wearable devices in the next years, it would be interesting to propose more **system** oriented MEC-related works, resulting in or driven by real data traces.

Many open challenges have to do with security/privacy aspects and machine learning. Here we do not analyze those aspects, because they have not been studied in light of MEC deployments. However, the interested reader could find more

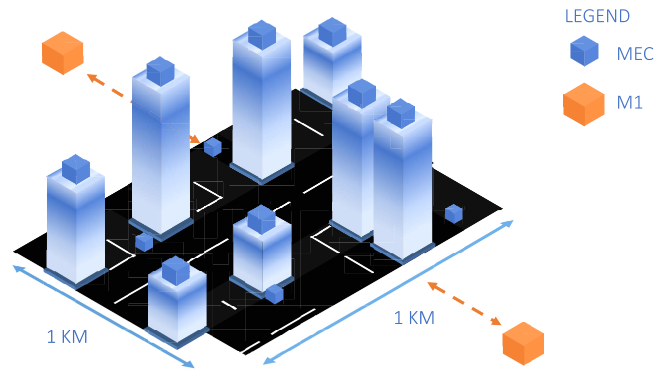

Fig. 9. MEC deployment scenario in a smart city district.

on those aspects in recently appeared surveys, e.g., [111], [114], [210], [211].

V. The Case of Smart Metropolitan Areas

The presence of connected devices is enhancing the cities and factories into smart entities with increasingly richer capabilities, evolving the concept from smart cities into a wider smart *metropolitan* area, which goes beyond the city itself since it includes a mix of areas where people leave and work, and also where services are produced and manufacturing happens. This allows for new communication and computing scenarios, e.g., for the interaction between (autonomous) vehicles and pedestrians, the dynamic management of the electrical resources and of AR/VR applications. To make these and other applications happen, it is of fundamental importance to actually guarantee the promised high data rates, high compute power and low latency that came with 5G systems. The MEC paradigm is pivotal in this framework.

In this section we highlight the features of the MEC deployment in a smart metropolitan environment, tackling the QoE requirements of citizens and workers, and the possible infrastructure bottlenecks, considering several verticals all together and a massive user presence. Fig. 9 shows a district in a smart metropolitan area. The cellular network covers an area of one square kilometer and consists of twelve 5G antennas deployed near to a corresponding MEC host, as described in [51]. According to the topology of access and transport networks proposed in [212], which is based on ITU recommendations [213], six 5G antennas (hence in our case six MEC hosts) are grouped under a single M1 access node, which is placed at an average distance of 10-20 km from the MEC hosts. Hence for each square kilometer, there will be two M1 dedicated nodes. Inside most of the biggest European cities the M1 access node would hence be placed outside of downtown. Every group of six M1 access nodes are connected to an M2 node, typically located 80-100 km away from the M1 node. However, here we do not consider M2 nodes and higher concentration nodes, whose distance from the user makes the propagation delay non-negligible.

A. Network Capabilities and Use Cases

Table V shows the values for network capabilities and requirements of MEC hosts, as suggested by 3GPP [39], and the corresponding values for computational capabilities, taken from [178]. In 1 km^2 the backhaul will offer a downlink (DL) capacity from the core of the network of 750 Gbps, distributed over two M1 nodes. The uplink (UL) will be in excess of 125 Gbps per square kilometer. Every MEC host can use, on average, at least 62.5 Gbps in DL and 10.41 Gbps in UL. With six gNBs per M1 node, these numbers correspond to the backhaul capacity of each gNB. These values are much higher than what can be offered by existing access network technologies, which therefore introduce a bottleneck for what concerns the actual speed observed by the users. For instance, the new standard for wireless communications 802.11ac will achieve a maximum throughput of 1.3 Gbps while the new 5G NR will achieve throughput up to a few Gbps [214].

Computational resources offered within the considered district are also quite powerful: for a MEC node located next to a gNB, it is possible to deploy a few servers (e.g., 16 servers), each with a few cores and GHz CPU rates (e.g., four cores at 3.4 GHz). For a MEC located the M1 node, the number of servers can grow much higher, e.g., 256 [178].

In the example portrayed in Fig. 9, MEC resources are exploited only by users located outside buildings, without considering indoor hot-spots [39]. We consider a highly dense metropolitan area, with up to 25,000 users connected in square kilometer [39], which is the order of magnitude of the population density in the biggest European capitals. In particular, we build an example based on four representative use cases: (*i*) vehicle collisions warnings with Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM) messages, (*ii*) video streaming and broadcasting, (*iii*) smart factories and (*iv*) VR/AR. Table VI summarizes the per-use-case requirements, taken from [126], [191], [151], [175], [206], [215], and [216]. In our example, for sake of simplicity, we assume that each user generates one task at a time for each request.

Thanks to new connectivity possibilities, nowadays it is possible to improve road safety leveraging CAM and DENM messages delivered from or to a vehicle, with collisions avoidance algorithms processed at MEC nodes. The goal of these messages is to check if a collision can eventually happen and, in case, to warn nearby vehicles. Partially due to the small payload of messages, DL and UL minimum requirements for successful deliveries of CAM and DENM messages are quite small: 4kbps for DL and 70 kbps for UL, while latency should range between 10 ms and a maximum of 100 ms [215], which corresponds to the generation rate of CAM messages (10 Hz) [126]. Instead, video streaming imposes more stringent and powerful requirements per service request: 70 Mbps for DL, 25 Mbps for UL and a maximum latency of 10ms [151]. However, the considered bandwidth requirements have been taken from measures from video broadcasting case scenarios (such as video sharing during a concert or sport live event inside a crowded stadium) and therefore, depending on the actual video streaming, requirements may vary.

Smart factory requirements vary as well, depending on the use case. For instance, controlling mobile robots need at least a data rate of 10 Mbps and a latency of 10-100 ms, while motion control devices (such as packing machines)

TABLE V
NETWORK CAPABILITIES AND COMPUTATIONAL RESOURCES

	DL bandwidth	UL bandwidth	Compute power (machines $\times$ cores $\times$ CPU speed)
MEC host at gNB site	62.5 Gbps	10.41 Gbps	16 $\times$ 4 $\times$ 3.4 GHz
MEC at M1 access node	$>$ 375 Gbps	$>$ 62.5 Gbps	256 $\times$ 4 $\times$ 3.4 GHz

TABLE VI
PER-USE-CASE REQUIREMENTS

Use case	DL bandwidth	UL bandwidth	RTT	Compute power
Vehicle collisions warning	4 kbps	70 kbps	10-100 ms	up to 43 $\times 10^6$ cycles/task, minimum of 217600 tasks/s
Video streaming	70 Mbps	25 Mbps	10 ms	up to 1 $\times 10^9$ cycles/task, minimum of 2176 tasks/s
Smart Factories	$>$ 1 Mbps	$>$ 1 Mbps	1-100 ms	up to 1.936 $\times 10^9$ cycles/task, minimum of 114 tasks/s
AR/VR	4K 2D 100 Mbps 24K 3D 2-5 Gbps	6,45 Mbps	30 ms 10 ms	up to 40 $\times 10^9$ cycles/task, minimum of 6 tasks/s

require less bandwidth (at least 1 Mbps) but a stricter latency (1 ms) [206].

The AR/VR use case imposes very stringent requirements: for a basic experience with 4K resolution of 2D videos, according to [175] [182], the DL bandwidth needed is 100 Mbps with an RTT of $<$ 30 ms, whilst for a full immersive experience (24K and 3D) the requirements go up to 2-5 Gbps of DL bandwidth, with less than 10 ms of round-trip latency [175]. A possible UL value for AR/VR applications is 6,45 Mbps [217].

B. Bottlenecks and Scalability

Taking a look at bandwidth capabilities, it is already possible to draw several conclusions: for instance, in the worst case scenario, with all 25000 users connected at the same time, the bandwidth for a fully immersive AR/VR experience cannot be guaranteed at all by gNBs and backhaul. However, if up to 30% of the users are connected (7500 users), the full backhaul can provide enough bandwidth for the most basic AR/VR applications. Still, with 1.5 Gbps available for UL at the antenna, the single gNB cannot serve more than about 15 AR/VR users with the least acceptable quality, so no more than 180 users can be served by the 12 gNBs present in the district. While fulfilling the enormous requirements for a fully interconnected VR is still utopia, as also highlighted in [178], other verticals instead could benefits from the MEC presence in both terms on bandwidth, computing capabilities and latency. For instance, considering the same capabilities, the infrastructure is able to serve 22 video streamers per gNB, and a total of 264 users. In the other considered cases, the numbers grow very much. Indeed, a single gNB supports up to 1500 IIoT devices in UL, enough for the average number of devices envisioned for a smart factory (according to [206], the number of IIoT devices could range from 2 up to 10000 per km^2).

Furthermore, for vehicle collision warnings, the numbers are even higher: more than 375,000 parallel communication sessions are supported between the infrastructure and vehicles! Anyway, it is important to highlight that for smart factories and vehicle communications, MEC improved bandwidth capabilities are not as important as maintaining a reliability of 99.999%, otherwise catastrophic situations could occur (such as collisions between vehicles or IIoT malfunctioning devices, with enormous economic damages for factories).

As a remark, real case scenarios are much more complex: gNBs and MEC nodes should be able to sustain *several* verticals at the same time, therefore providing bandwidth resources for AR/VR but also for vehicle collision warnings and smart factories, etc. When bandwidth becomes the bottleneck, a solution might be offered by deploying several gNBs per MEC node, although it would incur further CAPEX/OPEX costs. Furthermore, we need to consider computational and latency limits as well.

Looking at the latency requirements, they should always be guaranteed apart for the processing tasks delay. Indeed, thanks to the dense antenna and MEC deployment in the considered area, propagation delays on air, copper and/or fiber are negligible (below half millisecond per 100 km [218]), while processing packet delay at a MEC host running at medium load is of the order of one μs [219]. However, congestion must be avoided, which boils down to under-utilize links and MEC hosts. For instance, typical queueing and computing architectures do not experience the buildup of large queues if used below 65-75% of their capacity, depending on the distribution of task arrivals [220]. It is therefore safe to count only on two thirds or at most three quarters of the transmission capacity in the deployment (the same holds for computational capacity). For instance, this means that, at least from a point of view of available transmission resources, one should not accommodate more than $\approx$ 5250 AR/VR streams and 1050 IIoT devices.

Moreover, the numbers reported above need to be modified in case the compute power becomes the bottleneck. Specifically, focusing on computational resources, Figure 10 shows the maximum number of users U that can be served at the same time as a function of the processing cycles required to serve a task. We obtained these values by dividing the total computing power (CPU cycles per second) C of a MEC node by the ratio between the processing cycles (P_C) required for a task and the task deadline D (i.e., the number of CPU cycles per second required to serve a task):

$$U = \frac{C}{P_C} D \tag{1}$$

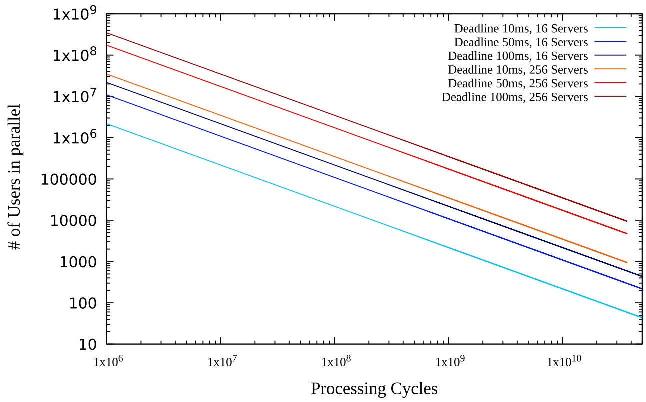

Fig. 10. Number of users satisfied in parallel, according to different demands of processing cycles per task. Each user generates 1 task at a time for request.

In this example, we consider various numbers of servers, processing cycles and task deadlines. Specifically, the deadlines reported in Fig. 10, i.e., 10ms, 50ms and 100ms, indicate latency values that cannot be exceeded to provide optimal QoE ranging from video streaming services to vehicles collisions warnings, whereas server provisioning per MEC node consists in typical values of 16 or 256 servers. The heavier the computing tasks, the less the number of users that can be served at the same time, with an inverse proportional relation between the two quantities (which appears as linear in the log-log scale used in the figure). For instance, exploiting light computing services, it is possible to serve more than 100,000 users respecting the deadline of 10ms. It is the case of vehicles collisions warnings, where short messages size is mandatory in order to achieve faster information spreading across the vehicles. Further, it is possible to notice that both bandwidth and computing capabilities do not represent a clear bottleneck for this service, which therefore depends on the proximity of MEC resources [129] for latency issues. Instead, video streaming requires higher computational loads: considering a face recognition use case, a single task can require up to 1 billion cycles. So, taking into account a latency of $\approx$ 10 ms, the infrastructure can support a maximum of 3000 users. The same happens for IIoT devices: according to [216] a critical task requires up to 1.93×10^9 cycles, hence $\approx$ 1200 devices can be satisfied at the same time, a scenario one order of magnitude higher than the one described by 5G ACIA [206]. Finally, for a fully interconnected experience, AR/VR hits computation limits before bandwidth ones, and only 100 users can be served in the smart city district of our example in the case of 37 billion cycles per tasks for massive VR applications (as highlighted in [178]).

Summarizing, apart for the vehicle use case, all other considered cases show important limitations to support a massive presence of users (the scenario evaluated considered up to 25000 people) in terms of computing capabilities. Furthermore, it is important to highlight two main aspects: (*i*) proposed computing capabilities are clearly an over-provisioning exercise for very edge nodes (for instance a Nokia edge datacenter supports up to five servers) (*ii*) to avoid uncontrollable response times depending on the distribution of jobs arrival [220], MEC host capacity should not be exploited more than 65-70%. This means that, compared with the numbers described in Figure 10, for the same amount of processing cycles, servers should be used to serve no more than 65-70% of the nominal capacity, in terms of number of users. Therefore, if we consider a reduction of 50% of the server capabilities, which are then used only for the 65-70% of their full capacity, the numbers are quite different. For instance, now video streaming is supported for up to 1050 users in parallel on the same MEC host, which is still more than what can be served with the available bandwidth. However, in the smart factory use case, up to 140 IIoT can be supported, and for the fully interconnected VR case, the MEC node can serve only 25 AR/VR devices. These numbers, compared to the ones obtained by considering the bandwidth, show that computation power can soon become the bottleneck. Furthermore, it is important to highlight that computing resources could be also shared among different slices. Therefore, as for the bandwidth capabilities case, less computing resources could be available causing a reduction of served users or an increasing of processing delays [95].

The number of tasks per second supported by one MEC host varies depending on the number of users connected at the same time and it is inversely proportional with the processing cycles required. Table VI shows the values in the worst case scenario, when tasks require more processing cycles:

for vehicle collision warnings, it is possible to sustain a minimum of 217600 tasks/s, while the number drops to 2176 tasks/s for the video streaming case. Furthermore, one MEC host supports up to 114 tasks/s in the smart factory case and 6 tasks/s in the AR/VR case. As for the previous cases, if we consider a reduction MEC servers by 50% which are able to serve only for the 65-70% of their nominal capacity, supported tasks/s decrease accordingly (e.g., 40 tasks/s for a smart factory and a minimum of 4 tasks/s for AR/VR).

C. Required Density of MEC Hosts and Its Cost

Now we want to consider how many MEC hosts are required to support a given number of users connected simultaneously, considering different use cases for bandwidth constraints or computing power limits, and evaluate the associated cost (Fig. 13).

Fig. 11 shows the density of MEC hosts needed as a function of users density, according to the **bandwidth** requirements of different verticals, which are listed in Table VI. Here we considered one MEC host per gNB site. Firstly, it is important to notice the behavior of the warning collision messages use case: the bandwidth required for both DL and UL is so small that only a single MEC host can sustain the full range of user densities considered here (up to 25000 users/km^2). Instead, for all the other use cases, the curve of required MEC hosts has a staircase shape. While the smart factory use case needs six MEC to sustain up to 25000 users in a square km, therefore remaining under the threshold of 12 MEC hosts per square km proposed earlier, all the other cases (video and AR/VR and mixed traffic) require extra deployments of MEC hosts. If 25000 users connect at the same time in a square km area,

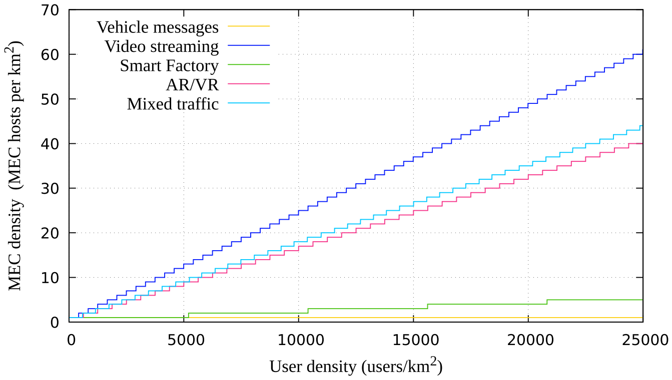

Fig. 11. Density of MEC hosts required in different use cases, according to bandwidth requirements (refer to table V and VI for parameters).

and everyone leverages on AR/VR services, at least 41 MEC hosts are needed in that area to guarantee enough bandwidth for both DL and UL, while 62 MEC hosts are needed for the video use case. The curve labeled as *Mixed traffic* represents a scenario in which a mix of all four use cases is present. Specifically, we design the mix of traffic according to Cisco[28] and Ericsson[29] traffic forecasts for the following few years: 70% video traffic, 15% car traffic, 10% smart factory and 5% AR/VR. For 25000 mixed users/km^2, more than 40 MEC hosts per square km are required (i.e., more than the 12 proposed earlier). In all of our considered cases, UL imposes tighter constraints, so that the bandwidth bottleneck is imposed by the aggregate UL traffic.

Furthermore, we study the required density of MEC hosts according to **computing power** needs of the users. In Fig. 12, we consider presence of the MEC hosts at gNB sites as well as M1 nodes, with a ratio of one M1 node every six MEC hosts. To avoid unrealistic deployment scenarios, we limited the deployment of new MEC hosts up to 96 (hence $\approx$ 1 MEC hosts per 100 m^2). This justifies the curves ending before reaching the maximum population density considered (i.e., curves stop where the capacity of 96 MEC hosts per square km, and the associated M1 nodes, has been reached). In the figure, we notice that, while the vehicle warning messages use case again requires very low computing capabilities (only 1 MEC host per square km), the other cases behave differently. The AR/VR use case saturates the MEC computing capacity (up to 96 MEC hosts) with in less than 2000 served users. Instead, video streaming, smart factory and the mixed traffic scenarios are all able to sustain a traffic of 25000 users/km^2 or more. More specifically, up to 82 MEC hosts plus 16 M1 nodes are needed in the area of one square km to serve the mixed traffic case. The numbers fgo down to 60 MEC hosts and 12 M1 nodes for the smart factory case, and further down to 22 MEC hosts plus 6 M1 nodes for video streaming. It is possible to notice that, apart the video streaming use case, all other cases require more MEC nodes for providing computing capacity than what they need for bandwidth. This shows that computing represents the real bottleneck in most of the cases.

Fig. 13 shows the infrastructure CAPEX needed to deploy MEC nodes in a square km, as a function of user density. This time we consider both bandwidth and computing power requirements. In the figure, we consider a cost of $\approx$ 2000 USD per deployed server plus other CAPEX costs such as new base stations deployments, civil works and small cell equipment ($\approx$ 94000 USD) [221]. We also consider the cost of M1 nodes, for each of which we use the extracted CAPEX cost of deploying a 256-server datacenter ($\approx$ 1.5 million USD) evaluated with AWS cost calculator.[30]

From the figure, it is possible to see that, again, the cost to sustain the vehicle warning messages case remains steady, due to the low bandwidth and computing requirements. Video streaming and smart factory have the same long term behavior: they are able to sustain the whole population while reaching a final cost of 25 million USD per km^2.

In the mixed traffic scenario, the cost to sustain as many as 25000 users per square km is higher, summing up to 34.8 million USD per km^2. In the AR/VR case, we observe the highest costs (up to slightly more than 35.8 million USD with less than 2000 users/km^2). However, it is interesting to notice how the AR/VR use case alone increases the infrastructure CAPEX costs, therefore giving a new design constraint to infrastructure providers. This behavior could be viewed especially in the mixed traffic scenario, where it contributes only 5% to the overall traffic.

Summary: We showed how in a smart metropolitan context both the bandwidth and the computing capabilities, even when quite powerful, require the deployment of new MEC nodes, exceeding therefore the threshold of 12 MEC nodes. Furthermore, we showed that especially the computing capabilities represent a clear bottleneck for the network infrastructure. This however doesn't mean that a heterogeneous smart metropolis is not possible: while advanced AR/VR is still much beyond the nowadays network capabilities, connected cars exchanging simple collision warning messages together with video streaming and smart factories might coexist together, placing a first step towards the path of a fully interconnected metropolitan area.

D. Lessons Learned

The analysis of network deployment in a smart metropolitan area highlights some lessons learned and points out some problems that need to be addressed:

First, the computational capabilities of the MEC deployment should be carefully considered as a function of the expected verticals operated in the served area, since different verticals (e.g., video streaming, smart factories or AR/VR) have very different requirements [222]. Second, it seems very impractical, from a pure cost perspective, to scale up a typical 5G use case for a big crowd of devices. For instance, according to Intel,[31] in the next future a single autonomous driving car

[28]https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[29]https://www.ericsson.com/en/mobility-report/reports/november-2019/mobile-traffic-by-application-category

[30]https://awstcocalculator.com

[31]https://newsroom.intel.com/editorials/krzanich-the-future-of-automated-driving/

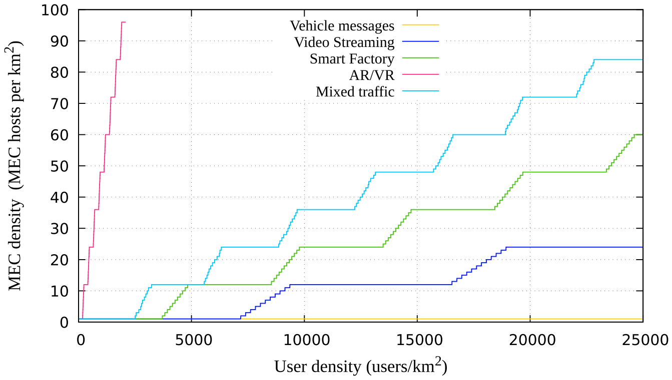

Fig. 12. Density of MEC hosts which are required, with their associated M1 nodes, to serve different use cases based on bandwidth requirements (see Table V and VI for parameters).

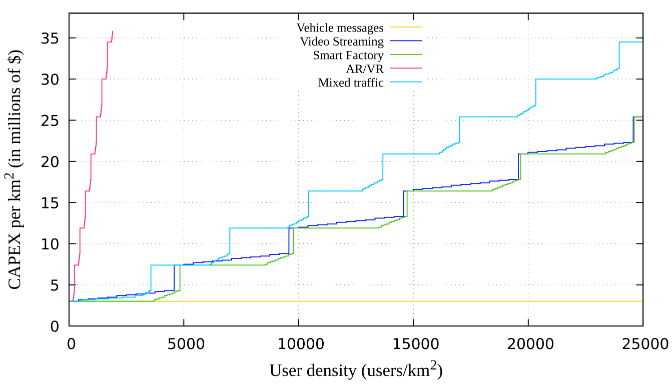

Fig. 13. Infrastructure CAPEX cost per km^2.

will generate up to 4 terabyte of data per day, which would require either very powerful MEC hosts or very dense and expensive MEC deployments just to serve a few tens of cars per unit area. Third, future MEC host solutions should consider leveraging GPUs instead of CPUs for entertainment use cases such as gaming or AR/VR, and leveraging ML per forecasting task arrivals, allowing to allocate/scale resources in advance. In addition, they could leverage smart computation offloading in order to avoid unnecessary offload to MEC hosts.

In addition to the above points, our simple examples show that the MEC deployment in a urban district or a metropolitan area can require high densities, which incurs logistic problems and constraints, and hence requires carefully designed deployment plans which account for presence of natural or artificial obstacles while guaranteeing uniform reachability and access to bandwidth and computational resources. The use of renewables would be desirable, in accordance to recently proposed energy-awareness efforts, e.g., to follow the guidelines of the European Green Deal agenda[32] or of the Microsoft Green Computing initiative.[33]

VI. Conclusion

The MEC is a crucial paradigm for the future Internet infrastructure. It will be fundamental for the successful delivery of 5G services and for their evolution beyond 5G systems, as well as for the support of the increasingly heterogeneous connected devices. This survey has firstly offered a comprehensive overview of MEC standardization and steering efforts. Then, we have reviewed works that leverage how to enforce the provisioning of MEC resources to offer effective services. This includes works on computation offloading, (flexible) MEC resources deployment and agile migration of VNFs. Eventually, we have surveyed several MEC use cases within the principal 5G-PPP vertical industries (i.e., automotive, smart city, media, manufacturing, and eHealthcare). For all these topics, we have described the lessons learned and open research challenges. Finally, we commented on a possible MEC deployment scenario in a smart metropolitan area, for which we have showed that several use cases require very different MEC deployments. From our study, it emerges a serious risk that MEC does not scale sufficiently well, in terms of logistics and costs, so to hinder the development of verticals that require an edge computing infrastructure to perform adequately, so as to penetrate the market.

[32] https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal_en
[33] https://blogs.microsoft.com/green/

References

[1] P. M. Mell and T. Grance, "The NIST definition of cloud computing," NIST, Gaithersburg, MD, USA, Rep. SP 800–145, 2011.

[2] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2015.

[3] ETSI, "Mobile edge computing—A key technology towards 5G," ETSI MEC ISG, Sophia Antipolis, France, White Paper, Sep. 2015.

[4] ETSI, "Mobile edge computing (MEC); Deployment of mobile edge computing in an NFV environment," ETSI MEC ISG, Sophia Antipolis, France, Rep. GR MEC 017 V1.1.1 (2018-02), Feb. 2018.

[5] S. Yi, C. Li, and Q. Li, "A survey of fog computing: Concepts, applications and issues," in *Proc. Workshop Mobile Big Data*, New York, NY, USA, 2015, pp. 37–42. [Online]. Available: http://doi.acm.org/10.1145/2757384.2757397

[6] M. Satyanarayanan, V. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct.–Dec. 2009. [Online]. Available: https://www.microsoft.com/en-us/research/publication/the-case-for-vm-based-cloudlets-in-mobile-computing/

[7] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017.

[8] J. Moura and D. Hutchison, "Game theory for multi-access edge computing: Survey, use cases, and future trends," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 260–288, 1st Quart., 2019.

[9] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.

[10] W. Yu *et al.*, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.

[11] A. C. Baktir, A. Ozgovde, and C. Ersoy, "How can edge computing benefit from software-defined networking: A survey, use cases, and future directions," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2359–2391, 4th Quart., 2017.

[12] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, Fog *et al.*: A survey and analysis of security threats and challenges," *Future Gener. Comput. Syst.*, vol. 78, no. 2, pp. 680–698, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16305635

[13] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of Things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, 4th Quart., 2018.

[14] S. N. Shirazi, A. Gouglidis, A. Farshad, and D. Hutchison, "The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2586–2595, Nov. 2017.

[15] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[16] S. Shahzadi, M. Iqbal, T. Dagiuklas, and Z. U. Qayyum, "Multi-access edge computing: Open issues, challenges and future perspectives," *J. Cloud Comput.*, vol. 6, no. 1, p. 30, Dec. 2017. [Online]. Available: https://doi.org/10.1186/s13677-017-0097-9

[17] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[18] Q. Pham *et al.*, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116974–117017, 2020.

[19] M. Patel *et al.*, "Mobile-edge computing introductory technical white paper," Mobile-Edge Computing (MEC) Industry Initiative, Sophia Antipolis, France, White Paper, 2014.

[20] ETSI, "Multi-access edge computing (MEC); Framework and reference architecture," ETSI MEC ISG, Sophia Antipolis, France, Rep. GS MEC 003 V2.1.1 (2019-01), Jan. 2019.

[21] M. S. K. Ha, "Openstack++ for cloudlet deployment," document CMU-CS-15-123, School Comput. Sci., Carnegie Mellon University, Pittsburgh, PA, USA, Aug. 2015.

[22] ETSI, "Network functions virtualisation (NFV); Management and orchestration," ETSI MEC ISG, Sophia Antipolis, France, Rep. GS NFV-MAN 001 V1.1.1 (2014-12), Dec. 2014.

[23] V. Sciancalepore, F. Giust, K. Samdanis, and Z. Yousaf, "A double-tier MEC-NFV architecture: Design and optimisation," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Oct. 2016, pp. 1–6.

[24] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[25] ETSI, "MEC in 5G networks," ETSI MEC ISG, Sophia Antipolis, France, White Paper, Jun. 2018.

[26] 3GPP, "3GPP technical specification group services and system aspects; system architecture for the 5G system," 3GPP, Sophia Antipolis, France, Rep. TS 123 501, 2019.

[27] ETSI, "Multi-access edge computing (MEC); Support for network slicing," ETSI MEC ISG, Sophia Antipolis, France, Rep. GR MEC 024, Nov. 2019.

[28] A. Ksentini and P. A. Frangoudis, "Toward slicing-enabled multi-access edge computing in 5G," *IEEE Netw.*, vol. 34, no. 2, pp. 99–105, Mar./Apr. 2020.

[29] L. Cominardi, T. Deiss, M. Filippou, V. Sciancalepore, F. Giust, and D. Sabella, "MEC support for network slicing: Status and limitations from a standardization viewpoint," *IEEE Commun. Stand. Mag.*, vol. 4, no. 2, pp. 22–30, Jun. 2020.

[30] 3GPP, "Study on enhancement of support for edge computing in 5G core network (5GC)," 3GPP, Sophia Antipolis, France, Rep. SA2 TR 23.748, 2020.

[31] 3GPP, "Architecture for enabling edge applications;," 3GPP, Sophia Antipolis, France, Rep. SA6 TS23.558, 2020.

[32] 3GPP, "Study on application architecture for enabling edge applications," 3GPP, Sophia Antipolis, France, Rep. TR 23.758, 2019.

[33] L. M. Contreras and C. J. Bernardos, "Overview of architectural alternatives for the integration of ETSI MEC environments from different administrative domains," *Electronics*, vol. 9, no. 9, p. 1392, 2020.

[34] G. Castellano, A. Manzalini, and F. Risso, "A disaggregated MEC architecture enabling open services and novel business models," in *Proc. IEEE Conf. Netw. Softw. (Netsoft)*, Jun. 2019, pp. 178–182.

[35] ETSI, "Cloud RAN and MEC: A perfect pairing," ETSI MEC ISG, Sophia Antipolis, France, White Paper, 2018. [Online]. Available: www.etsi.org

[36] ETSI, "MEC in an enterprise setting: A solution outline," ETSI MEC ISG, Sophia Antipolis, France, White Paper, 2018.

[37] ETSI, "Multi-access edge computing (MEC); Study on MEC support for V2X use cases," ETSI MEC ISG, Sophia Antipolis, France, Rep. GR MEC 022 V2.1.1 (2018-09), 2018.

[38] "5G at the edge," 5G Americas, Bellevue, WA, USA, White Paper, Oct. 2019.

[39] 3rd Generation Partnership Project, *Service Requirements for Next Generation New Services and Markets, Version 16.5.0*, 3GPP Standard TS 22.261, Sep. 2018.

[40] O-RAN ALLIANCE, "O-RAN use cases and deployment scenarios white paper," O-RAN ALLIANCE, Alfter, Germany, Rep., Feb. 2020.

[41] O-RAN ALLIANCE, "O-RAN towards an open and smart RAN white paper," O-RAN ALLIANCE, Alfter, Germany, Rep., Oct. 2019.

[42] G. Brown, "Mobile edge computing use cases & deployment options prepared by," Juniper, Sunnyvale, CA, USA, White Paper, Jul. 2016.

[43] V. Frascolla *et al.*, "5G-Miedge: Design, standardization and deployment of 5G phase II technologies: MEC and mmwaves joint development for Tokyo 2020 olympic games," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Sep. 2017, pp. 54–59.

[44] S. Arora, P. A. Frangoudis, and A. Ksentini, "Exposing radio network information in a MEC-in-NFV environment: The RNISaaS concept," in *Proc. IEEE Conf. Netw. Softw. (NetSoft)*, 2019, pp. 306–310.

[45] ETSI, "Multi-access edge computing (MEC); Radio network information API slicing," ETSI MEC ISG, Sophia Antipolis, France, Rep. 012 V2.1.1, Dec. 2019.

[46] L. Zanzi, F. Giust, and V. Sciancalepore, "M^2EC: A multi-tenant resource orchestration in multi-access edge computing systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.

[47] A. Huang, N. Nikaein, T. Stenbock, A. Ksentini, and C. Bonnet, "Low latency MEC framework for SDN-based LTE/LTE-A networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[48] T. Taleb, P. A. Frangoudis, I. Benkacem, and A. Ksentini, "CDN slicing over a multi-domain edge cloud," *IEEE Trans. Mobile Comput.*, vol. 19, no. 9, pp. 2010–2027, Sep. 2020.

[49] X. Lyu *et al.*, "Selective offloading in mobile edge computing for the green Internet of Things," *IEEE Netw.*, vol. 32, no. 1, pp. 54–60, Jan. 2018.

[50] D. Sabella *et al.*, "Edge computing: From standard to actual infrastructure deployment and software development," Intel, Mountain View, CA, USA, Rep., 2019.

[51] J. Martín-Pérez, L. Cominardi, C. J. Bernardos, A. de la Oliva, and A. Azcorra, "Modeling mobile edge computing deployments for low latency multimedia services," *IEEE Trans. Broadcast.*, vol. 65, no. 2, pp. 464–474, Jun. 2019.

[52] M. Syamkumar, P. Barford, and R. Durairajan, "Deployment characteristics of 'the edge' in mobile edge computing," in *Proc. Workshop Mobile Edge Commun.*, New York, NY, USA, 2018, pp. 43–49. [Online]. Available: http://doi.acm.org/10.1145/3229556.3229557

[53] M. Bouet and V. Conan, "Geo-partitioning of MEC resources," in *Proc. Workshop Mobile Edge Commun.*, New York, NY, USA, 2017, pp. 43–48. [Online]. Available: http://doi.acm.org/10.1145/3098208.3098216

[54] P. Vitello, A. Capponi, C. Fiandrino, G. Cantelmo, and D. Kliazovich, "The impact of human mobility on edge data center deployment in urban environments," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[55] N. Kherraf, H. A. Alameddine, S. Sharafeddine, C. M. Assi, and A. Ghrayeb, "Optimized provisioning of edge computing resources with heterogeneous workload in IoT networks," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 2, pp. 459–474, Jun. 2019.

[56] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 725–737, Oct.–Dec. 2017.

[57] M. C. Filippou, D. Sabella, and V. Riccobene, "Flexible MEC service consumption through edge host zoning in 5G networks," 2019. [Online]. Available: http://arxiv.org/abs/1903.01794.

[58] G. Castellano, F. Esposito, and F. Risso, "A distributed orchestration algorithm for edge computing resources with guarantees," *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 2548–2556. [Online]. Available: http://par.nsf.gov/biblio/10082130

[59] F. Slim, F. Guillemin, A. Gravey, and Y. Hadjadj-Aoul, "Towards a dynamic adaptive placement of virtual network functions under ONAP," in *Proc. IEEE Conf. Netw. Funct. Virtualization Softw. Defined Netw. (NFV-SDN)*, Nov. 2017, pp. 210–215.

[60] S. Salsano *et al.*, "Toward superfluid deployment of virtual functions: Exploiting mobile edge computing for video streaming," in *Proc. 29th Int. Teletraffic Congr. (ITC)*, vol. 2, Sep. 2017, pp. 48–53.

[61] L. Yala, P. A. Frangoudis, and A. Ksentini, "Latency and availability driven VNF placement in a MEC-NFV environment," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7.

[62] K. Poularakis, J. Llorca, A. M. Tulino, I. Taylor, and L. Tassiulas, "Joint service placement and request routing in multi-cell mobile edge computing networks," *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2019. [Online]. Available: http://dx.doi.org/10.1109/INFOCOM.2019.8737385

[63] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12,

pp. 8402–8413, Dec. 2013.

[64] R. Bruschi, F. Davoli, P. Lago, C. Lombardo, and J. F. Pajo, "Personal services placement and low-latency migration in edge computing environments," in *Proc. IEEE Conf. Netw. Funct. Virtualization Softw. Defined Netw. (NFV-SDN)*, Nov. 2018, pp. 1–6.

[65] V. Farhadi *et al.*, "Service placement and request scheduling for data-intensive applications in edge clouds," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2019, pp. 1279–1287.

[66] S. Yang, F. Li, S. Trajanovski, X. Chen, Y. Wang, and X. Fu, "Delay-aware virtual network function placement and routing in edge clouds," *IEEE Trans. Mobile Comput.*, early access, Sep. 19, 2019, doi: 10.1109/TMC.2019.2942306.

[67] T. He, H. Khamfroush, S. Wang, T. L. Porta, and S. Stein, "It's hard to share: Joint service placement and request scheduling in edge clouds with sharable and non-sharable resources," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2018, pp. 365–375.

[68] B. P. Rimal, D. P. Van, and M. Maier, "Mobile edge computing empowered fiber-wireless access networks in the 5G era," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 192–200, Feb. 2017.

[69] C.-Y. Li *et al.*, "Mobile edge computing platform deployment in 4G LTE networks: A middlebox approach," in *Proc. USENIX Workshop Hot Topics Edge Comput. (HotEdge)*, Boston, MA, USA, 2018. [Online]. Available: https://www.usenix.org/conference/hotedge18/presentation/li

[70] A. van Kempen, T. Crivat, B. Trubert, D. Roy, and G. Pierre, "MEC-ConPaas: An experimental single-board based mobile edge cloud," in *Proc. 5th IEEE Int. Conf. Mobile Cloud Comput. Services Eng. (MobileCloud)*, Apr. 2017, pp. 17–24.

[71] T. Taleb and A. Ksentini, "Follow me cloud: Interworking federated clouds and distributed mobile networks," *IEEE Netw.*, vol. 27, no. 5, pp. 12–19, Sep./Oct. 2013.

[72] T. Taleb, A. Ksentini, and P. A. Frangoudis, "Follow-me cloud: When cloud services follow mobile users," *IEEE Trans. Cloud Comput.*, vol. 7, no. 2, pp. 369–382, Apr.–Jun. 2019.

[73] R. A. Addad, D. L. C. Dutra, M. Bagaa, T. Taleb, and H. Flinck, "Towards a fast service migration in 5G," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Oct. 2018, pp. 1–6.

[74] A. Machen, S. Wang, K. K. Leung, B. J. Ko, and T. Salonidis, "Live service migration in mobile edge clouds," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 140–147, Feb. 2018.

[75] A. Aissioui, A. Ksentini, A. M. Gueroui, and T. Taleb, "On enabling 5G automotive systems using follow me edge-cloud concept," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5302–5316, Jun. 2018.

[76] I. Farris, T. Taleb, M. Bagaa, and H. Flick, "Optimizing service replication for mobile delay-sensitive applications in 5G edge network," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[77] I. Farris, T. Taleb, A. Iera, and H. Flinck, "Lightweight service replication for ultra-short latency applications in mobile edge networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[78] A. K. Sangaiah, D. V. Medhane, T. Han, M. S. Hossain, and G. Muhammad, "Enforcing position-based confidentiality with machine learning paradigm through mobile edge computing in real-time industrial informatics," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4189–4196, Jul. 2019.

[79] G. A. Carella, M. Pauls, T. Magedanz, M. Cilloni, P. Bellavista, and L. Foschini, "Prototyping NFV-based multi-access edge computing in 5G ready networks with open baton," *Proc. IEEE Conf. Netw. Softw. Softw. Sustain. Hyper-Connected World en Route 5G (NetSoft)*, 2017, pp. 1–4.

[80] B. I. Ismail *et al.*, "Evaluation of docker as edge computing platform," in *Proc. IEEE Conf. Open Syst. (ICOS)*, Aug. 2015, pp. 130–135.

[81] G. Avino, M. Malinverno, F. Malandrino, C. Casetti, and C. F. Chiasserini, "Characterizing docker overhead in mobile edge computing scenarios," in *Proc. Workshop Hot Topics Container Netw. Netw. Syst.*, New York, NY, USA, 2017, pp. 30–35. [Online]. Available: http://doi.acm.org/10.1145/3094405.3094411

[82] S. Wang, R. Urgaonkar, M. Zafer, T. He, K. Chan, and K. K. Leung, "Dynamic service migration in mobile edge computing based on Markov decision process," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1272–1288, Jun. 2019.

[83] T. V. Doan *et al.*, "Containers vs virtual machines: Choosing the right virtualization technology for mobile edge cloud," in *Proc. IEEE 2nd 5G World Forum (5GW)*, 2019, pp. 46–52.

[84] L. Ma, S. Yi, N. Carter, and Q. Li, "Efficient live migration of edge services leveraging container layered storage," *IEEE Trans. Mobile Comput.*, vol. 18, no. 9, pp. 2020–2033, Sep. 2019.

[85] K. Ha *et al.*, "Adaptive VM handoff across cloudlets," School Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, USA, Rep. CMU-CS-15-113, 2015.

[86] P. J. Braun, S. Pandi, R. Schmoll, and F. H. P. Fitzek, "On the study and deployment of mobile edge cloud for tactile internet using a 5G gaming application," in *Proc. 14th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2017, pp. 154–159.

[87] R. Cziva, C. Anagnostopoulos, and D. P. Pezaros, "Dynamic, latency-optimal VNF placement at the network edge," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2018, pp. 693–701.

[88] H. A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi, and C. Assi, "Dynamic task offloading and scheduling for low-latency IoT services in multi-access edge computing," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 668–682, Mar. 2019.

[89] X. He, R. Jin, and H. Dai, "Peace: Privacy-preserving and cost-efficient task offloading for mobile-edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1814–1824, Mar. 2020.

[90] J. Yan, S. Bi, Y. J. Zhang, and M. Tao, "Optimal task offloading and resource allocation in mobile-edge computing with inter-user task dependency," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 235–250, Jan. 2020.

[91] X. Meng, W. Wang, Y. Wang, V. K. N. Lau, and Z. Zhang, "Closed-form delay-optimal computation offloading in mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4653–4667, Oct. 2019.

[92] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, "Optimal energy management policies for energy harvesting sensor nodes," *IEEE Trans. Wireless Commun.*, vol. 9, no. 4, pp. 1326–1336, Apr. 2010.

[93] L. Huang and M. J. Neely, "Utility optimal scheduling in energy-harvesting networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1117–1130, Aug. 2013.

[94] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019.

[95] Z. Liang, Y. Liu, T. Lok, and K. Huang, "Multiuser computation offloading and downloading for edge computing with virtualization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4298–4311, Sep. 2019.

[96] S. Jošilo and G. Dán, "Computation offloading scheduling for periodic tasks in mobile edge computing," *IEEE/ACM Trans. Netw.*, vol. 28, no. 2, pp. 667–680, Apr. 2020.

[97] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 4924–4938, Aug. 2017.

[98] Y. Cheng, J. Zhang, L. Yang, C. Zhu, and H. Zhu, "Distributed green offloading and power optimization in virtualized small cell networks with mobile edge computing," *IEEE Trans. Green Commun. Netw.*, vol. 4, no. 1, pp. 69–82, Mar. 2020.

[99] J. Li, A. Wu, S. Chu, T. Liu, and F. Shu, "Mobile edge computing for task offloading in small-cell networks via belief propagation," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–6.

[100] M. Li, F. R. Yu, P. Si, and Y. Zhang, "Energy-efficient machine-to-machine (M2M) communications in virtualized cellular networks with mobile edge computing (MEC)," *IEEE Trans. Mobile Comput.*, vol. 18, no. 7, pp. 1541–1555, Jul. 2019.

[101] S. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Energy efficiency and delay tradeoff for wireless powered mobile-edge computing systems with multi-access schemes," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1855–1867, Mar. 2020.

[102] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[103] F. Malandrino and C. Chiasserini, "Getting the most out of your VNFs: Flexible assignment of service priorities in 5G," 2019. [Online]. Available: http://arxiv.org/abs/1904.00704.

[104] M. Hua, Y. Wang, C. Li, Y. Huang, and L. Yang, "UAV-aided mobile edge computing systems with one by one access scheme," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 3, pp. 664–678, Sep. 2019.

[105] Z. Yang, C. Pan, K. Wang, and M. Shikh-Bahaei, "Energy efficient resource allocation in UAV-enabled mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4576–4589, Sep. 2019.

[106] L. Zanzi *et al.*, "Evolving multi-access edge computing to support enhanced IoT deployments," *IEEE Commun. Stand. Mag.*, vol. 3, no. 2, pp. 26–34, Jun. 2019.

[107] P. L. Ventre *et al.*, "Segment routing: A comprehensive survey of research activities, standardization efforts and implementation results," 2020. [Online]. Available: arXiv:1904.03471.

[108] C. Cicconetti, M. Conti, A. Passarella, and D. Sabella, "Toward distributed computing environments with serverless solutions in edge systems," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 40–46, Mar. 2020.

[109] T. He, E. N. Ciftcioglu, S. Wang, and K. S. Chan, "Location privacy in mobile edge clouds: A chaff-based approach," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2625–2636, Nov. 2017.

[110] X. He, J. Liu, R. Jin, and H. Dai, "Privacy-aware offloading in mobile-edge computing," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, 2017, pp. 1–6.

[111] R. Khan, P. Kumar, D. N. K. Jayakody, and M. Liyanage, "A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 196–248, 1st Quart., 2020.

[112] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[113] S. Wang *et al.*, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[114] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019.

[115] G. Vallero, D. Renga, M. Meo, and M. A. Marsan, "Greener RAN operation through machine learning," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 3, pp. 896–908, Sep. 2019.

[116] D. Zhang *et al.*, "Near-optimal and truthful online auction for computation offloading in green edge-computing systems," *IEEE Trans. Mobile Comput.*, vol. 19, no. 4, pp. 880–893, Apr. 2020.

[117] V. Passas, N. Makris, V. Miliotis, and T. Korakis, "Pricing based MEC resource allocation for 5G heterogeneous network access," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[118] R. Ricart-Sanchez, P. Malagon, J. M. Alcaraz-Calero, and Q. Wang, "P4-net FPGA-based network slicing solution for 5G MEC architectures," in *Proc. ACM/IEEE Symp. Archit. Netw. Commun. Syst. (ANCS)*, 2019, pp. 1–2.

[119] R. A. Uzcategui, A. J. De Sucre, and G. Acosta-Marum, "Wave: A tutorial," *IEEE Commun. Mag.*, vol. 47, no. 5, pp. 126–133, May 2009.

[120] S. A. A. Shah, E. Ahmed, M. Imran, and S. Zeadally, "5G for vehicular communications," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 111–117, Jan. 2018.

[121] R. Soua, I. Turcanu, F. Adamsky, D. Fuhrer, and T. Engel, "Multi-access edge computing for vehicular networks: A position paper," *Proc. IEEE Globecom Workshops (GC Wkshps)*, 2019, pp. 1–6.

[122] M. Emara, M. C. Filippou, and D. Sabella, "MEC-assisted end-to-end latency evaluations for C-V2X communications," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2018, pp. 1–9.

[123] Q. Hu, C. Wu, X. Zhao, X. Chen, Y. Ji, and T. Yoshinaga, "Vehicular multi-access edge computing with licensed sub-6 GHz, IEEE 802.11p and mmWave," *IEEE Access*, vol. 6, pp. 1995–2004, 2018.

[124] "Toward fully connected vehicles: Edge computing for advanced automotive communications," 5GAA, Munich, Germany, White Paper, Feb. 2019.

[125] Q.-H. Nguyen, M. Morold, K. David, and F. Dressler, "Car-to-pedestrian communication with MEC-support for adaptive safety of vulnerable road users," *Comput. Commun.*, vol. 150, pp. 83–93, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0140366419304360

[126] M. Malinverno, J. Mangues-Bafalluy, C. E. Casetti, C. F. Chiasserini, M. Requena-Esteso, and J. Baranda, "An edge-based framework for enhanced road safety of connected cars," *IEEE Access*, vol. 8, pp. 58018–58031, 2020.

[127] M. Malinverno, G. Avino, C. Casetti, C. F. Chiasserini, F. Malandrino, and S. Scarpina, "MEC-based collision avoidance for vehicles and vulnerable users," 2019. [Online]. Available: arXiv:1911.05299.

[128] M. Malinverno, G. Avino, C. Casetti, C. F. Chiasserini, F. Malandrino, and S. Scarpina, "Performance analysis of C-V2I-based automotive collision avoidance," in *Proc. IEEE 19th Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, 2018, pp. 1–9.

[129] G. Avino *et al.*, "A MEC-based extended virtual sensing for automotive services," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 4, pp. 1450–1463, Dec. 2019.

[130] J. Heinovski and F. Dressler, "Platoon formation: Optimized car to platoon assignment strategies and protocols," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Dec. 2018, pp. 1–8.

[131] U. Montanaro, S. Fallah, M. Dianati, D. Oxtoby, T. Mizutani, and A. Mouzakitis, "Cloud-assisted distributed control system architecture for platooning," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 1258–1265.

[132] R. Huang, B. Chang, Y. Tsai, and Y. Liang, "Mobile edge computing-based vehicular cloud of cooperative adaptive driving for platooning autonomous self driving," in *Proc. IEEE 7th Int. Symp. Cloud Service Comput. (SC2)*, Nov. 2017, pp. 32–39.

[133] T. Cui, X. Fan, C. Cao, and Q. Chen, "Minimum cost offloading decision strategy for collaborative task execution of platooning assisted by MEC," in *Communications and Networking*, X. Liu, D. Cheng, and L. Jinfeng, Eds. Cham, Switzerland: Springer Int. Publ., 2019, pp. 104–115.

[134] A. Virdis, G. Nardini, and G. Stea, "A framework for MEC-enabled platooning," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshop (WCNCW)*, Apr. 2019, pp. 1–6.

[135] C. Quadri, V. Mancuso, M. G. A. Marsan, and G. P. Rossi, "Platooning on the edge," in *Proc. 23nd Int. ACM Conf. Model. Anal. Simulat. Wireless Mobile Syst.*, New York, NY, USA, 2020, pp. 1–10.

[136] K. Zhang, Y. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Optimal delay constrained offloading for vehicular edge computing networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–6.

[137] Z. Xiao *et al.*, "Vehicular task offloading via heat-aware MEC cooperation using game-theoretic method," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2038–2052, Mar. 2020.

[138] M. Gerla, "Vehicular cloud computing," in *Proc. 11th Annu. Mediterranean Ad Hoc Netw. Workshop (Med-Hoc-Net)*, Jun. 2012, pp. 152–155.

[139] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," 2013. [Online]. Available: http://arxiv.org/abs/1308.6208.

[140] R. Copeland *et al.*, "Technology assessment for mission-critical services on automotive virtual edge communicator (AVEC)," in *Proc. 21st Conf. Innov. Clouds Internet Netw. Workshops (ICIN)*, Feb. 2018, pp. 1–8.

[141] F. Dressler, P. Handle, and C. Sommer, "Towards a vehicular cloud—Using parked vehicles as a temporary network and storage infrastructure," in *Proc. ACM Int. Workshop Wireless Mobile Technol. Smart Cities*, New York, NY, USA, 2014, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/2633661.2633671

[142] F. Hagenauer, C. Sommer, T. Higuchi, O. Altintas, and F. Dressler, "Vehicular micro clouds as virtual edge servers for efficient data collection," in *Proc. 2nd ACM Int. Workshop Smart Auton. Connected Veh. Syst. Services*, New York, NY, USA, 2017, pp. 31–35. [Online]. Available: http://doi.acm.org/10.1145/3131944.3133937

[143] F. Hagenauer, C. Sommer, T. Higuchi, O. Altintas, and F. Dressler, "Vehicular micro cloud in action: On gateway selection and gateway handovers," *Ad Hoc Netw.*, vol. 78, pp. 73–83, Sep. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870518302464

[144] F. Dressler, G. S. Pannu, F. Hagenauer, M. Gerla, T. Higuchi, and O. Altintas, "Virtual edge computing using vehicular micro clouds," in *Proc. Int. Conf. Comput. Netw. Commun. (ICNC)*, Feb. 2019, pp. 537–541.

[145] A. Ndikumana and C. S. Hong, "Self-driving car meets multi-access edge computing for deep learning-based caching," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2019, pp. 49–54.

[146] A. Ndikumana, N. H. Tran, D. H. Kim, K. T. Kim, and C. S. Hong, "Deep learning based caching for self-driving cars in multi-access edge computing," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 4, 2020, doi: 10.1109/TITS.2020.2976572.

[147] S. Garg, A. Singh, S. Batra, N. Kumar, and L. T. Yang, "UAV-empowered edge computing environment for cyber-threat detection in smart vehicles," *IEEE Netw.*, vol. 32, no. 3, pp. 42–51, May 2018.

[148] S. Husain, A. Kunz, A. Prasad, K. Samdanis, and J. Song, "Mobile edge computing with network resource slicing for Internet-of-Things," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Feb. 2018, pp. 1–6.

[149] D. Sabella, A. Vaillant, P. Kuure, U. Rauschenbach, and F. Giust, "Mobile-edge computing architecture: The role of MEC in the Internet of Things," *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 84–91, Oct. 2016.

[150] F. Cirillo, D. Gómez, L. Diez, I. E. Maestro, T. B. J. Gilbert, and R. Akhavan, "Smart city IoT services creation through large-scale collaboration," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5267–5275, Jun. 2020.

[151] C. Colman-Meixner *et al.*, "Deploying a novel 5G-enabled architecture on city infrastructure for ultra-high definition and immersive media production and broadcasting," *IEEE Trans. Broadcast.*, vol. 65, no. 2, pp. 392–403, Jun. 2019.

[152] C. Vallati, A. Virdis, E. Mingozzi, and G. Stea, "mobile-edge computing come home connecting things in future smart homes using LTE device-to-device communications," *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 77–83, Oct. 2016.

[153] Y. Deng, Z. Chen, X. Yao, S. Hassan, and J. Wu, "Task scheduling for smart city applications based on multi-server mobile edge computing," *IEEE Access*, vol. 7, pp. 14410–14421, 2019.

[154] Y. He, F. R. Yu, N. Zhao, V. C. M. Leung, and H. Yin, "Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 31–37, Dec. 2017.

[155] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE Netw.*, vol. 33, no. 2, pp. 111–117, Mar. 2019.

[156] L. Zhao, J. Wang, J. Liu, and N. Kato, "Routing for crowd management in Smart Cities: A Deep Reinforcement learning perspective," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 88–93, 2019.

[157] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, and H. Flinck, "Mobile edge computing potential in making cities smarter," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 38–43, Mar. 2017.

[158] B. Wang, M. Li, X. Jin, and C. Guo, "A reliable IoT edge computing trust management mechanism for smart cities," *IEEE Access*, vol. 8, pp. 46373–46399, 2020.

[159] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in P2P networks," in *Proc. 12th Int. Conf. World Wide Web*, New York, NY, USA, 2003, p. 640–651. [Online]. Available: https://doi.org/10.1145/775152.775242

[160] X. Fan, L. Liu, M. Li, and Z. Su, "Grouptrust: Dependable trust management," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 4, pp. 1076–1090, Apr. 2017.

[161] D. Wang, B. Bai, K. Lei, W. Zhao, Y. Yang, and Z. Han, "Enhancing information security via physical layer approaches in heterogeneous IoT with multiple access mobile edge computing in smart city," *IEEE Access*, vol. 7, pp. 54508–54521, 2019.

[162] M. A. Rahman, M. M. Rashid, M. S. Hossain, E. Hassanain, M. F. Alhamid, and M. Guizani, "Blockchain and IoT-based cognitive edge framework for sharing economy services in a smart city," *IEEE Access*, vol. 7, pp. 18611–18621, 2019.

[163] M. Gheisari, Q. Pham, M. Alazab, X. Zhang, C. Fernández-Campusano, and G. Srivastava, "ECA: An edge computing architecture for privacy-preserving in IoT-based smart city," *IEEE Access*, vol. 7, pp. 155779–155786, 2019.

[164] G. Ananthanarayanan *et al.*, "Real-time video analytics: The killer app for edge computing," *Computer*, vol. 50, no. 10, pp. 58–67, 2017.

[165] T. X. Tran and D. Pompili, "Adaptive bitrate video caching and processing in mobile-edge computing networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 9, pp. 1965–1978, Sep. 2019.

[166] T. X. Tran, D. V. Le, G. Yue, and D. Pompili, "Cooperative hierarchical caching and request scheduling in a cloud radio access network," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2729–2743, Dec. 2018.

[167] Y. Hung, C. Wang, and R. Hwang, "Optimizing social welfare of live video streaming services in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 4, pp. 922–934, Apr. 2020.

[168] A. Mehrabi, M. Siekkinen, and A. Ylä-ääski, "Edge computing assisted adaptive mobile video streaming," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 787–800, Apr. 2019.

[169] N. Bouten, S. Latré, J. Famaey, W. Van Leekwijck, and F. De Turck, "In-network quality optimization for adaptive video streaming services," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2281–2293, Dec. 2014.

[170] C. Long, Y. Cao, T. Jiang, and Q. Zhang, "Edge computing framework for cooperative video processing in multimedia IoT systems," *IEEE Trans. Multimedia*, vol. 20, no. 5, pp. 1126–1139, May 2018.

[171] S. Yang, Y. Tseng, C. Huang, and W. Lin, "Multi-access edge computing enhanced video streaming: Proof-of-concept implementation and prediction/QoE models," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1888–1902, Feb. 2019.

[172] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 695–708, Jan. 2019.

[173] A. Martin, R. Viola, M. Zorrilla, J. Flórez, P. Angueira, and J. Montalbán, "MEC for fair, reliable and efficient media streaming in mobile networks," *IEEE Trans. Broadcast.*, vol. 66, no. 2, pp. 264–278, Jun. 2020.

[174] C. Ge and N. Wang, "Real-time QoE estimation of DASH-based mobile video applications through edge computing," in *Proc. IEEE Conf. Comput. Commun. Workshops (IEEE INFOCOM WKSHPS)*, Apr. 2018, pp. 766–771.

[175] L. Huawei Technologies Company, "Virtual reality/augmented reality white paper," China Academy of Information and Communications Technology (CAICT), Beijing, China, Rep., Dec. 2017.

[176] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Netw.*, vol. 32, no. 2, pp. 78–84, Mar. 2018.

[177] J. Liu, G. Shou, Y. Liu, Y. Hu, and Z. Guo, "Performance evaluation of integrated multi-access edge computing and fiber-wireless access networks," *IEEE Access*, vol. 6, pp. 30269–30279, 2018.

[178] E. Bastug, M. Bennis, M. Medard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 110–117, Jun. 2017.

[179] X. Yang *et al.*, "Communication-constrained mobile edge computing systems for wireless virtual reality: Scheduling and tradeoff," *IEEE Access*, vol. 6, pp. 16665–16677, 2018.

[180] Y. Liu, J. Liu, A. Argyriou, and S. Ci, "MEC-assisted panoramic VR video streaming over millimeter wave mobile networks," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1302–1316, May 2019.

[181] Y. Sun, Z. Chen, M. Tao, and H. Liu, "Communications, caching, and computing for mobile virtual reality: Modeling and tradeoff," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7573–7586, Nov. 2019.

[182] S. Mangiante, G. Klas, A. Navon, Z. GuanHua, J. Ran, and M. D. Silva, "VR is on the edge: How to deliver 360° videos in mobile networks," in *Proc. Workshop Virtual Reality Augmented Reality Netw.*, New York, NY, USA, 2017, pp. 30–35. [Online]. Available: http://doi.acm.org/10.1145/3097895.3097901

[183] 3GPP, "Study on communication for automation in vertical domains," 3GPP, Sophia Antipolis, France, Rep. TR 22.804, 2018.

[184] M. Hermann, T. Pentek, and B. Otto, "Design principles for industries 4.0 scenarios," in *Proc. 49th Hawaii Int. Conf. Syst. Sci. (HICSS)*, Jan. 2016, pp. 3928–3937.

[185] P. Schulz *et al.*, "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 70–78, Feb. 2017.

[186] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.

[187] N. Dao, Y. Lee, S. Cho, E. Kim, K. Chung, and C. Keum, "Multi-tier multi-access edge computing: The role for the fourth industrial revolution," in *Proc. Int. Conf. Inf. Commun. Technol. Convergence (ICTC)*, Oct. 2017, pp. 1280–1282.

[188] X. Li, J. Wan, H. Dai, M. Imran, M. Xia, and A. Celesti, "A hybrid computing solution and resource scheduling strategy for edge computing in smart manufacturing," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4225–4234, Jul. 2019.

[189] E. E. Ugwuanyi, S. Ghosh, M. Iqbal, and T. Dagiuklas, "Reliable resource provisioning using bankers' deadlock avoidance algorithm in MEC for Industrial IoT," *IEEE Access*, vol. 6, pp. 43327–43335, 2018.

[190] S. Luo, Y. Wen, W. Xu, and D. Puthal, "Adaptive task offloading auction for industrial CPS in mobile edge computing," *IEEE Access*, vol. 7, pp. 169055–169065, 2019.

[191] C. Hsu, Y. Hsu, and H. Wei, "Energy-efficient and reliable MEC offloading for heterogeneous industrial IoT networks," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2019, pp. 384–388.

[192] T. N. Gia, M. Jiang, A. Rahmani, T. Westerlund, P. Liljeberg, and H. Tenhunen, "Fog computing in healthcare Internet of Things: A case study on ECG feature extraction," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. Ubiquitous Comput. Commun. Dependable Auton. Secure Comput. Pervasive Intell. Comput.*, Oct. 2015, pp. 356–363.

[193] A. Abdellatif, A. Emam, C.-F. Chiasserini, A. Mohamed, A. Jaoua, and R. Ward, "Edge-based compression and classification for smart healthcare systems: Concept, implementation and evaluation," *Expert Syst. Appl.*, vol. 117, pp. 1–14, Mar. 2019.

[194] A. A. Abdellatif, A. Mohamed, C. F. Chiasserini, M. Tlili, and A. Erbad, "Edge computing for smart health: Context-aware approaches, opportunities, and challenges," *IEEE Netw.*, vol. 33, no. 3, pp. 196–203, May 2019.
[195] G. Muhammad, M. F. Alhamid, and X. Long, "Computing and processing on the edge: Smart pathology detection for connected healthcare," *IEEE Netw.*, vol. 33, no. 6, pp. 44–49, Nov./Dec. 2019.
[196] P. Pace, G. Aloi, R. Gravina, G. Caliciuri, G. Fortino, and A. Liotta, "An edge-based architecture to support efficient applications for healthcare industry 4.0," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 481–489, Jan. 2019.
[197] A. Islam and S. Y. Shin, "BHMUS: Blockchain based secure outdoor health monitoring scheme using UAV in smart city," in *Proc. 7th Int. Conf. Inf. Commun. Technol. (ICoICT)*, 2019, pp. 1–6.
[198] M. Chen, W. Li, H. Yixue, Y. Qian, and I. Humar, "Edge cognitive computing based smart healthcare system," *Future Gener. Comput. Syst.*, vol. 86, pp. 403–411, Sep. 2018.
[199] T. Muhammed, R. Mehmood, A. Albeshri, and I. Katib, "UbeHealth: A personalized ubiquitous cloud and edge-enabled networked healthcare system for smart cities," *IEEE Access*, vol. 6, pp. 32258–32285, 2018.
[200] X. Li, X. Huang, C. Li, R. Yu, and L. Shu, "EdgeCare: Leveraging edge computing for collaborative data management in mobile healthcare systems," *IEEE Access*, vol. 7, pp. 22011–22025, 2019.
[201] F. Giannone, P. A. Frangoudis, A. Ksentini, and L. Valcarenghi, "Orchestrating heterogeneous MEC-based applications for connected vehicles," *Comput. Netw.*, vol. 180, Oct. 2020, Art. no. 107402. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128620301997
[202] F. Giust *et al.*, "Multi-access edge computing: The driver behind the wheel of 5G-connected cars," *IEEE Commun. Stand. Mag.*, vol. 2, no. 3, pp. 66–73, Sep. 2018.
[203] D. Grewe, M. Wagner, M. Arumaithurai, I. Psaras, and D. Kutscher, "Information-centric mobile edge computing for connected vehicle environments: Challenges and research directions," in *Proc. Workshop Mobile Edge Commun.*, New York, NY, USA, 2017, pp. 7–12. [Online]. Available: http://doi.acm.org/10.1145/3098208.3098210
[204] W. Zhang, L. Li, N. Zhang, T. Han, and S. Wang, "Air-ground integrated mobile edge networks: A survey," *IEEE Access*, vol. 8, pp. 125998–126018, 2020.
[205] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu, "Edge computing in industrial Internet of things: Architecture, advances and challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2462–2488, 4th Quart., 2020.
[206] 5G Alliance for Connected Industries and Automation, "5G for connected industries and automation," 5G ACIA, Frankfurt, Germany, Rep., Feb. 2019.
[207] 5G Alliance for Connected Industries and Automation, "5G for automation in industry: Primary use cases, functions and service requirements," 5G ACIA, Frankfurt, Germany, Rep., Jul. 2019.
[208] K. Antevski, C. J. Bernardos, L. Cominardi, A. de la Oliva, and A. Mourad, "On the integration of NFV and MEC technologies: Architecture analysis and benefits for edge robotics," *Comput. Netw.*, vol. 175, Jul. 2020, Art. no. 107274. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128620300797
[209] K. Antevski, M. Groshev, G. Baldoni, and C. J. Bernardos, "DLT federation for edge robotics," 2020. [Online]. Available: arXiv:2010.01977.
[210] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 869–904, 2nd Quart., 2020.
[211] T. K. Rodrigues, K. Suto, H. Nishiyama, J. Liu, and N. Kato, "Machine learning meets computation and communication control in evolving edge and cloud: Challenges and future perspective," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 38–67, 1st Quart., 2020.
[212] L. Cominardi, L. M. Contreras, C. J. Bcrnardos, and I. Berberana, "Understanding QoS applicability in 5G transport networks," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2018, pp. 1–5.
[213] ITU-T, "Consideration on 5G transport network reference architecture and bandwidth requirements," document Study Group 15 Contribution 0462, International Telecommunication Union–Telecommunication Standardization Sector (ITU-T), Geneva, Switzerland, Feb. 2017.
[214] N. Khalid and O. B. Akan, "Experimental throughput analysis of low-THz MIMO communication channel in 5G wireless networks," *IEEE Wireless Commun. Lett.*, vol. 5, no. 6, pp. 616–619, Dec. 2016.
[215] Z. Amjad, A. Sikora, B. Hilt, and J. Lauffenburger, "Low latency V2X applications and network requirements: Performance evaluation," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2018, pp. 220–225.
[216] H. Guo, J. Liu, and J. Zhang, "Efficient computation offloading for multi-access edge computing in 5G HetNets," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–6.
[217] W. Zhang, B. Han, and P. Hui, "On the networking challenges of mobile augmented reality," in *Proc. Workshop Virtual Reality Augmented Reality Netw.*, New York, NY, USA, 2017, pp. 24–29. [Online]. Available: http://doi.acm.org/10.1145/3097895.3097900
[218] V. Herminghaus and A. Scriba, *Storage Management in Data Centers: Understanding, Exploiting, Tuning, and Troubleshooting Veritas Storage Foundation*. Berlin, Germany: Springer, 2009.
[219] T. Zhang, L. Linguaglossa, P. Giaccone, L. Iannone, and J. Roberts, "Performance benchmarking of state-of-the-art software switches for NFV," 2020. [Online]. Available: arXiv:2003.13489.
[220] L. Kleinrock, "Time-shared systems: A theoretical treatment," *J. ACM*, vol. 14, no. 2, pp. 242–261, Apr. 1967. [Online]. Available: http://doi.acm.org/10.1145/321386.321388
[221] E. J. Oughton and Z. Frias, "The cost, coverage and rollout implications of 5G infrastructure in britain," *Telecommun. Policy*, vol. 42, no. 8, pp. 636–652, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0308596117302781
[222] F. Malandrino, C. Chiasserini, G. Avino, M. Malinverno, and S. Kirkpatrick, "From megabits to CPU ticks: Enriching a demand trace in the age of MEC," *IEEE Trans. Big Data*, vol. 6, no. 1, pp. 43–50, Mar. 2020.

Francesco Spinelli received the M.Sc. degree in communications and computer networks engineering from the Politecnico di Torino, Italy, in 2018. He is currently pursuing the Ph.D. degree with IMDEA Networks Institute, Spain, and with the University Carlos III of Madrid. He was a Research Engineer with Telecom Paris, France. His interests include MEC, machine learning, and renewable-energy systems.

Vincenzo Mancuso (Member, IEEE) received the Ph.D. degree from the University of Palermo, Italy, in 2005. He was with INRIA, France, Rice University, USA, and the University of Palermo. He is a Research Associate Professor with IMDEA Networks, Madrid, Spain. He was a recipient of a Ramon y Cajal research grant of the Spanish Ministry of Science and Innovation. His research focus is on analysis, design, and experimental evaluation of opportunistic wireless architectures and mobile broadband services.